# Intermodal Passenger Flows on London's Public Transport Network

## Automated Inference of Full Passenger Journeys Using Fare-Transaction and Vehicle-Location Data

by

Jason B. Gordon

B.A., University of California, Berkeley

Submitted to the Department of Civil and Environmental Engineering
and the Department of Urban Studies and Planning
in partial fulfillment of the requirements for the degrees of

Master of Science in Transportation
and
Master in City Planning
at the
Massachusetts Institute of Technology
September 2012



Signature of Author . . . . . .
Department of Civil and Environmental Engineering
Department of Urban Studies and Planning
August 10, 2012

Certified by . . . . . .
Nigel H.M. Wilson
Professor of Civil and Environmental Engineering
Thesis Supervisor

Certified by . . . . . .
Harilaos Koutsopoulos
Professor of Transport Science, Royal Institute of Technology
Thesis Supervisor

Certified by . . . . . .
John P. Attanucci
Lecturer & Research Associate, Department of Civil and Environmental Engineering
Thesis Supervisor

Accepted by . . . . . .
Alan Berger
Professor of Urban Studies and Planning
Chair, Master in City Planning Committee

Accepted by . . . . . .
Heidi M. Nepf
Professor of Civil and Environmental Engineering
Chair, Departmental Committee for Graduate Students

# Intermodal Passenger Flows on London's Public Transport Network

## Automated Inference of Full Passenger Journeys Using Fare-Transaction and Vehicle-Location Data

by
Jason B. Gordon

Submitted to the Department of Civil and Environmental Engineering
and the Department of Urban Studies and Planning
on August 10, 2012
in partial fulfillment of the requirements for the degrees of
Master of Science in Transportation
and
Master in City Planning

ABSTRACT

Urban public transport providers have historically planned and managed their networks and services with limited knowledge of their customers' travel patterns. While ticket gates and bus fareboxes yield counts of passenger activity in specific stations and vehicles, the relationships between these transactions—the origins, interchanges, and destinations of individual passengers—have typically been acquired only through costly and therefore small and infrequent rider surveys. Building upon recent work on the utilization of automated fare-collection and vehicle-location systems for passenger-behavior analysis, this thesis presents methods for inferring the full journeys of all riders on a large public transport network.

Using complete daily sets of data from London's Oyster farecard and iBus vehicle-location system, boarding and alighting times and locations are inferred for individual bus passengers, interchanges are inferred between passenger trips of various public modes, and full-journey origin–interchange–destination matrices are constructed, which include the estimated flows of non-farecard passengers. The outputs are validated against surveys and traditional origin–destination matrices, and the software implementation demonstrates that the procedure is efficient enough to be performed daily, enabling transport providers to observe travel behavior on all services at all times.

Thesis Supervisor: Nigel H.M. Wilson
Title: Professor of Civil and Environmental Engineering

Thesis Supervisor: Harilaos Koutsopoulos
Title: Professor of Transport Science, Royal Institute of Technology

Thesis Supervisor: John P. Attanucci
Title: Lecturer & Research Associate, Department of Civil and Environmental Engineering

# Acknowledgements

The research that culminated in this thesis evolved over a three-year period, during which I was extremely fortunate to have the guidance of multiple faculty members. Nigel Wilson and John Attanucci guided this work from its inception. Their intuitive knowledge of transit, standard of academic excellence, and complementary analytical styles guaranteed stimulating and fruitful discussion at each weekly meeting. Haris Koutsopoulos brought a similarly valuable perspective to the latter half of the project: his expertise and ingenuity strengthened the methodologies of Chapters 3 and 4 while providing the guidance to overcome the challenges addressed in Chapter 5. Haris and Nigel's detailed reviews of the manuscript greatly strengthened it—their interest and close attention are sincerely appreciated. Jinhua Zhao played a key role in guiding the early stages of this work, and the vision that he instilled was in mind throughout the project. Fred Salvucci kindly reviewed the manuscript as well. His policy insights were invaluable as always, and I'll miss his thoughtful weekly provisioning of yucca donuts.

This work would not have been possible without the support of Transport for London (TfL). Shashi Verma, Lauren Sager Weinstein, and John Barry provided the invaluable opportunity to work with and learn from their exceedingly talented teams, while their own feedback and support for the project were truly encouraging. Andrew Gaitskell answered countless arcane technical questions and extracted mountains of data. James McNair explored the algorithms and the code itself, lending a welcome (and clever) second perspective in what had previously been a solitary endeavor. Malcolm Fairhurst and Tony Richardson presented an opportunity to test this research with a real-world problem, while

Duncan Horne, Mark Roberts, Carol Smales, Mark Butlin, David Warner, Liam Henderson, Geoffrey Maunder, Dale Campbell, and Tim Cooper shared expert insights about all aspects of TfL and its data.

The staff of London Buses were equally instrumental in this project. Steve Robinson shared his unparalleled iBus expertise and, along with Nigel Hardy, Lisa Labrousse, Aidan Daly, and George Marcar, generously and patiently answered complex technical questions. Steve and Kevin Field tediously extracted large volumes of data, Annelies de Koning created indispensable data extraction tools, and Bob Blitz and Fergus McGhee shared their expert knowledge of the bus network. Alex Phillips and Keith Gardner developed a user-requirements framework which strengthened and refined the tools developed in this research. Special thanks to Rosa McShane for her friendship and hospitality, and for always handling logistic and bureaucratic issues.

Additional support was provided by the Department of Civil and Environmental Engineering through a Schoettler fellowship and a teaching assistantship, in which I had the rewarding experience of teaching with and learning from George Kocur. Tomer Toledo gave valuable feedback on the scaling problem, while Ginny Siggia and Kris Kipp guided me expertly through all manner of institutional hurdles.

This project grew from an initial collaboration with Albert Ng, whose talents and creativity are as valued as his friendship. Mike Frumin generously sacrificed hours of thesis-writing time to share his (locally) unparalleled knowledge of London's transport and information systems, while Winnie Wang and Yossi Ehrlich similarly lent their code, data, and methodological insights. Kevin Muhs exemplified the virtue of patience with his cheerful (and brave) de facto alpha testing, which yielded many valuable discoveries and suggestions (among many mutual tangential but welcome diversions). Gabriel Sánchez-Martínez shared his IPF source code, and helped vet scaling algorithms through numerous stimulating whiteboard discussions (which also benefitted from the insights of Sam Hickey, Nihit Jain, and Kari Hernandez). These and the other transit-lab students of 2010–2013 made the countless hours in the office far more enjoyable.

Lastly, thanks to my family for their encouragement and support, and to Dini for absolutely everything. The work in these pages was sustained by your love and sacrifice.

# Contents


LIST OF FIGURES . . . . . 11

LIST OF TABLES . . . . . 14

LIST OF ACRONYMS . . . . . 15

1 INTRODUCTION . . . . . 17

1.1 Motivation . . . . . 18

1.1.1 Full Journeys and Intermodal Travel. . . . . 19

1.1.2 The State of the Art . . . . . 19

1.1.3 Application to London . . . . . 20

1.2 Objectives . . . . . 21

1.3 Thesis Organization . . . . . 22

2 PUBLIC TRANSPORT IN LONDON . . . . . 23

2.1 London's Public Transport Network . . . . . 23

2.1.1 London Buses . . . . . 23

2.1.2 The London Underground. . . . . 24

2.1.3 National Rail . . . . . 24

2.1.4 London Rail . . . . . 25

2.2 Data Sources . . . . . 25

2.2.1 Oyster . . . . . 25

2.2.2 iBus . . . . . 29

2.2.3 Gateline and ETM . . . . . 29

2.2.4 Spatial Data . . . . . 30

3 BUS ORIGIN AND DESTINATION INFERENCE . . . . . 31
3.1 Previous Research . . . . . 32
3.1.1 Inferring Bus Origins Using AFC and AVL Data . . . . . 32
3.1.2 Inferring Bus Destinations Using AFC and AVL Data . . . . . 33
3.2 Origin Inference . . . . . 34
3.2.1 Exploratory Analysis of Input Data . . . . . 35
3.2.2 Origin-Inference Methodology . . . . . 38
3.2.3 Origin-Inference Results and Sensitivity Analysis . . . . . 40
3.3 Destination Inference . . . . . 43
3.3.1 Input Data . . . . . 44
3.3.2 Destination-Inference Methodology . . . . . 47
3.3.3 Destination-Inference Results and Sensitivity Analysis . . . . 52
3.4 Summary . . . . . 58
4 INTERCHANGE INFERENCE . . . . . 59
4.1 Concepts and Terminology . . . . . 60
4.2 Previous Research . . . . . 62
4.2.1 Discerning Activities Using AFC Data . . . . . 62
4.2.2 Interchange Inference Using AFC Data . . . . . 62
4.3 Methodology . . . . . 64
4.3.1 Binary Conditions . . . . . 66
4.3.2 Temporal Conditions . . . . . 68
4.3.3 Spatial Conditions . . . . . 70
4.4 Results and Sensitivity Analysis . . . . . 73
4.4.1 Sensitivity Analysis . . . . . 73
4.4.2 Results . . . . . 80
4.4.3 Comparison with the London Travel Demand Survey . . . . 82
4.5 Summary . . . . . 83

5 FULL-JOURNEY MATRIX EXPANSION . . . . . 85
5.1 Problem Definition . . . . . 85
5.2 Previous Research . . . . . 88
5.2.1 Iterative Proportional Fitting on Closed Networks . . . . . 88
5.2.2 Application of IPF on London's Public Transport Network . . 90
5.3 Input Data . . . . . 94
5.4 Methodology . . . . . 95
5.4.1 Problem Definition Revisited . . . . . 95
5.4.2 Problem Formulation and Solution . . . . . 98
5.5 Example: Test Network . . . . . 101
5.5.1 Validation . . . . . 101
5.5.2 Comparison to Iterative Proportional Fitting . . . . . 105
5.6 Application to the London Network . . . . . 108
5.6.1 Results . . . . . 109
5.6.2 Validation Against Iterative Proportional Fitting on the London Network . . . . . 112
5.7 Summary . . . . . 114
6 IMPLEMENTATION . . . . . 115
6.1 Overview . . . . . 116
6.2 Process . . . . . 118
6.2.1 Step 1: Preprocessing and Nearest-Stop Calculation . . . . 118
6.2.2 Step 2: Origin Inference and Daily History Reconstruction . . . . . 120
6.2.3 Step 3: Destination and Interchange Inference . . . . . 122
6.2.4 Step 4: Full-Journey Construction . . . . . 123
6.2.5 Step 5: Full-journey Matrix Expansion . . . . . 124
6.3 Results and Performance . . . . . 126
6.4 Summary . . . . . 126

7 APPLICATIONS . . . . . 127
7.1 Bus Applications . . . . . 128
7.1.1 Boarding/Alighting/Flow Profiles and Route-Level OD Matrices . . . . . 128
7.1.2 Bus Passenger Travel Time and Distance . . . . . 130
7.2 Cross-Modal and Full-Journey Applications . . . . . 132
7.3 Observing Changes in Rider Behavior and Demand . . . . . 134
7.4 Summary . . . . . 136
8 CONCLUSION . . . . . 139
8.1 Summary and Findings . . . . . 139
8.2 Recommendations . . . . . 141
8.3 Future Research . . . . . 143
REFERENCES . . . . . 147

# List of Figures


2-1 Schematic travel-zone map of London rail services . . . . . .26
2-2 Geographic map of London rail services . . . . . .27
2-3 Underground, National Rail, and DLR services in Central London . . .28
3-1 Oyster, BCMS, and iBus data for a portion of a single bus route . . . . .36
3-2 Activity diagram of the origin-inference process . . . . . .39
3-3 Distribution of origin-inference error for all Oyster bus boardings . .42
3-4 Letter-designated bus stops near Elephant and Castle Tube station . . .45
3-5 Elephant and Castle stops S and T . . . . . .45
3-6 Stop and station locations in Greater London . . . . . .46
3-7 Activity diagram of the destination-inference process. . . . . .49
3-8 Example of destination inference. . . . . .50
3-9 Distribution of average out-of-system speeds . . . . . .53
3-10 Distribution of Euclidean distances between candidate alighting stops and next fare transaction . . . . . .55
4-1 Activity diagram of the interchange-inference process . . . . . .67
4-2 Two potential interchange locations yielding equal cumulative Euclidean distances . . . . . .71
4-3 Interchange boundaries for various circuity ratios. . . . . .72
4-4 Time between cardholders' successive journey stages . . . . . .74
4-5 Distribution of elapsed times between candidate bus-stop arrivals and subsequent bus boardings. . . . . .77

4-6 Distribution of circuity ratios for potential interchanges . . . . . .78
4-7 Honor Oak Park to Bond Street via London Bridge . . . . . .79
4-8 London Fields to Angel via Hackney or Liverpool Street . . . . . .79
4-9 Distribution of potential full-journey distances . . . . . .80
4-10 Distribution of inferred journey durations. . . . . .82
4-11 Full journeys per passenger per day. . . . . .83
4-12 Stages per journey . . . . . .83
5-1 The test transit network . . . . . .86
5-2 IPF contingency table for the test network . . . . . .89
5-3 Example of the control-total adjustment process . . . . . .93
5-4 Example of estimated passenger flow on the test network . . . . . .97
5-5 Formulation and results of the full-journey scaling process, as applied to the test network. . . . . 102
5-6 Convergence of scaled transaction-node flows toward control totals . . . . . 104
5-7 Comparison of actual and estimated itinerary flows. . . . . 104
5-8 Comparison of passenger flows for the test network . . . . . 107
5-9 Hourly control totals for Victoria Underground Station . . . . . 108
5-10 Histograms of scaling factors . . . . . 110
5-11 Comparison of sample (or seed) flows to scaled flows . . . . . 111
5-12 Distributions of OD-pair scaling factors on London's rail network . . 112
5-13 Comparison of full-journey scaling and IPF by OD pair . . . . . 113
6-1 Overview of processes, inputs, and outputs . . . . . 116
6-2 Class diagram of origin-, destination-, and interchange-inference package . . . . . 117
6-3 Activity diagram of origin-, destination-, and interchange-inference processes . . . . . 119
6-4 Activity diagram of full-journey matrix-expansion process . . . . . 125
7-1 Boarding/alighting/flow profile for Route 488 inbound . . . . . 129
7-2 Origin–destination matrix for Route 488 inbound . . . . . 129

7-3 Comparison of average journey-stage lengths: GLBPS vs. OD-inferred Oyster . . . . . 130
7-4 Comparison of average journey-stage lengths by bus route: GLBPS vs. OD-inferred Oyster . . . . . 131
7-5 Origins of full journeys ending at Oxford Circus station . . . . . 133
7-6 Frame from a time-lapse animation of a full day's inferred Oyster activity. . . . . 134
7-7 First origins of day for riders of route 205x . . . . . 135
7-8 Boarding/alighting/flow profile for route 488, before and after its extension to Dalston Junction . . . . . 137
7-9 Origin–destination matrix for the extended Route 488 inbound . . . 137

# List of Tables


3-1 Detailed origin-inference results, ten-day average . . . . . 41

3-2 Summary of origin-inference results . . . . . 43

3-3 Cumulative frequencies of observed inter-stage distances . . . . . 55

3-4 Destination-inference results . . . . . 57

4-1 Median elapsed times between Oyster cardholders' consecutive bus boarding transactions . . . . . 64

4-2 Tests applied during the interchange-inference process . . . . . 66

4-3 Interchange-inference results: ten-day average . . . . . 81

5-1 London Underground weekday time periods . . . . . 91

5-2 Satisfaction of farebox and station-gate control totals . . . . . 109

# List of Acronyms

| | |
|---|---|
| AFC | Automatic fare collection |
| APC | Automatic passenger counter |
| ATOC | Association of Train Operating Companies |
| AVL | Automatic vehicle location |
| BCMS | Bus Contract Management System |
| BODS | Bus Origin–Destination Survey |
| DLR | Docklands Light Railway |
| ELL | East London Line |
| ETM | Electronic ticket machine |
| GLBPS | Greater London Bus Passenger Survey |
| IPF | Iterative proportional fitting |
| LATS | London Area Travel Survey |
| LTDS | London Travel Demand Survey |
| MBTA | Massachusetts Bay Transportation Authority |
| MLE | Maximum likelihood estimate |
| NLC | National location code |
| OD | Origin–destination |
| ODX | Origin–destination–interchange |
| OSI | Out-of-station interchange |
| RODS | Rolling Origin–Destination Survey |
| TfL | Transport for London |

# Introduction 1

Urban public transport providers have historically planned and managed their networks and services with limited knowledge of their customers' travel patterns. Unlike airlines or interurban rail providers, who typically issue tickets specifying distinct origins, destinations, and transfer points—often in designated vehicles and seats—the nature of urban transit operations has necessitated fare-payment schemes which yield only aggregate ridership data, at resolutions no finer than that of the station gate or bus farebox.[1] Put simply, transit agencies have not been able to easily observe where their passengers travel.

Farebox and station-gate data are useful for describing ridership from the perspective of the transit system, but passenger-centric data, such as the origins, destinations, and transfer points of individual customers, have traditionally been collected only through costly surveys (which are therefore conducted infrequently using small rider samples). The growing adoption of automated data-collection systems, however, is providing transit operators with vast stores of disaggregate data, which can be manipulated to reveal travel information from a rider-focused perspective. But the distillation of passenger-centric information is not trivial, as most automated data-collection systems—namely automated fare collection (AFC), automated vehicle location (AVL), and automated passenger coun-

1 High peak-period passenger volumes, high but often variable service frequencies, and dense, complex networks have led most transit agencies to adopt flat or zonal fare policies, designed to expedite fare processing and maximize passenger throughput.

ter (APC)—have been designed for other purposes: their applicability to rider-perspective analysis is mostly serendipitous.[2]

This thesis builds upon recent work in the synthesis of passenger-centric public transit information and demonstrates the feasibility of processing the complete set of data for a large multi-modal transit provider in a way that is suitable for execution on a daily basis. Using London's transit network as an example, origins and destinations are inferred for bus journey stages, transfers between stages of various modes are inferred, and a multimodal origin–interchange–destination matrix of full journeys is constructed, including estimated flows for non-AFC passengers. Applications are presented at various spatial and temporal scales from both the customer and system perspectives, including applications for service planning, rider behavior analysis, and project impact analysis.

The remainder of this chapter elaborates on the motivation for this research, presents its specific objectives and the approach taken, and outlines the structure of this document.

## 1.1 MOTIVATION

Passenger-centric transit information has historically been costly to capture. Origin–destination (OD) information for an the individual bus route could be estimated at the stop level from manual boarding counts combined with a sample of rider surveys (Ben Akiva et al 1985) or, at the expense of accuracy, with boarding counts alone (Simon and Furth 1985, Mishalani et al 2011). As survey response rates continue to decline, however, increasing their cost and bias (Stopher 2008, qtd. in Simon 2010), AFC data (supplemented in many cases by AVL data) have been shown to provide similar OD information at larger scales and at lower cost (Park et al 2008). A similar approach can be taken for rail networks requiring both exit and entry transactions, since AFC transactions can be associated by card number (Gordillo 2006, Chan 2007).

2 AFC systems, which read magnetic-stripe or RFID "smart" cards containing transit passes or credit, were designed to streamline fare collection. AVL systems track vehicle locations for the onboard announcement of stop information and to provide control centers with real-time fleet-management capability. APC systems track aggregate boardings and alightings but are not directly linked to specific passengers or origin–destination pairs.

### 1.1.1 *Full Journeys and Intermodal Travel*

In addition to relating entry and exit records, AFC data can be used to associate all of a cardholder's daily transactions, enabling transit providers to more easily discern interchanges from activities in order to infer customers' true origins and destinations. While analysis of the resultant full journeys and daily travel histories can provide valuable information for transport planners and operators, interchanges themselves are worthy of study because of their influence on customers' travel decisions (Wardman and Hine 2000, Hine and Scott 2000, Guo et al 2007, Eom et al 2011) and their role in enabling intermodal travel (Institute of Logistics and Transport 2000, Seaborn et al 2009, Transport for London 2002 & 2009).

The acceptance of AFC cards on multiple modes allows the observation of intermodal transit journeys, which are becoming increasingly important to transit providers and policy makers as a means of improving mobility through the elimination of barriers to cross-modal travel. Describing London's public transport strategy at the turn of the twenty-first century, then-mayor Ken Livingstone argued that "[modal] integration is not just about making the public transport network more attractive to existing and potential passengers, it is also about how the transport system can contribute to the achievement of broader economic, social, and environmental objectives" (Transport for London 2001). The economic benefits of increased mobility are supported by the work of Krizek (2003) and by Anas (2007), who notes that time savings from chained trips tend to be reinvested in additional travel and activities, yielding an overall increase in utility. London's exceptionally large, dense transit network and the acceptance of the Oyster smart card on the region's several historically independent but now unified transit modes provides a valuable opportunity for intermodal integration and its monitoring through automatically collected data.

### 1.1.2 *The State of the Art*

Several scholars and practitioners have advanced the application of automatically collected transit data in the past decade. Building upon earlier work, Chu and Chapleau (2008) infer full passenger journeys for a bus network, then analyze multiple days of AFC data to estimate general activity types for cardholders' recurrent journeys (2010). Farzin (2008) auto-

mates the majority of the destination-inference process on a portion of a large bus network and constructs a zonally aggregated OD matrix. Munizaga et al (2011) automate the inference of origins and destinations for a large multi-modal network and, using a simple interchange-inference algorithm, construct a full-journey origin–destination matrix for AFC cardholders.

The synthesis and refinement of previous research, coupled with more robust and efficient processing algorithms, holds promise for the prospect of implementing passenger-centric data analysis in ways that can be applied by transit operators on a daily basis.

### 1.1.3 *Application to London*

Utsunomiya et al (2006) conclude that the market penetration of AFC cards is critical to the accuracy of passenger-centric analyses (2006). With penetration rates of over 90 percent on buses and 80 percent on the London Underground system, Oyster's six million daily bus boardings and five million station entries provide an exceptionally large sample from one of the world's foremost transit networks. Transport for London (TfL), the transportation agency for the Greater London metropolitan area, oversees the city's transit system and maintains complete databases of recent AFC, AVL, and ancillary transit data. TfL's data have been used to construct OD matrices for the gated Underground system (Gordillo 2006, Chan 2007), one of its ungated light rail systems (Henderson 2010), and for individual bus routes (Wang et al 2011). The data have also been used to track service reliability and to inform service-planning proposals (Chan 2007, Uniman et al 2010, Ehrlich 2010, Frumin 2010).

TfL's data sources have recently been used to assess the impacts of new capital projects (Ng 2011, Muhs 2012), fare policy changes (Jain 2011), and service-reliability metrics (Schil 2012).[3] The incorporation of previous London research with methodologies from other case studies presents an opportunity to extend traditional analyses to the observation of travel patterns before and after the introduction of new (or the alteration of existing) public transport services in London. And by developing an efficient and flexible way to repeat and extend such analyses, TfL can be

3 Muhs (2012) and Schil (2012) used the tools and algorithms developed in this thesis for part of their analyses.

equipped to internally analyze rider behavior during upcoming events such as the city's hosting of the 2012 Olympic and Paralympic Games and the launch of Crossrail service later in the decade.

## 1.2 OBJECTIVES

This research tests the hypothesis that a flow matrix of full intermodal passenger journeys can be estimated flexibly and efficiently using entire populations of automatically collected data from a large public transit network, in a way that can be performed by a transit provider on a daily basis. More specifically, this thesis seeks to fulfill the following objectives:

- *Infer boarding and alighting locations and times for bus journey stages*. Bus AFC transactions typically include service information and boarding times but lack spatial information, while alighting information is not recorded at all. Error-correction algorithms should be applied to ensure the highest possible origin- and destination-inference rates.

- *Infer interchanges between journey stages of any AFC-enabled mode*. Infer interchanges as accurately as possible, taking into account observed or inferred access and egress distances, observed bus headways, path circuity, and transfer time.

- *Generate an intermodal full-journey origin–interchange–destination matrix covering all modes in the transit network for which data are available*. Generate counts of each unique itinerary observed to have been taken by one (or more) passengers, where *itinerary* is defined as a unique combination of origin, destination, and transfer locations. Expansion factors, which are used to scale itinerary flows to account for unobserved travelers, should be estimated in such a way that the counts of their constituent journey stages satisfy the various control totals that are available for each transit mode.

- *Conduct all inference and expansion processes at the finest possible level of disaggregation*. Doing so will afford downstream processing and reporting tools the flexibility to choose a level of aggregation appropriate for a given task.

- *Allow parameters to be adjusted at runtime*. Bagchi and White (2003, 2004) note the importance of calibrating the parameters of rule-based processes in accordance with empirical observations or surveys, while Timmermans et al (2002) show that travel parameters vary across networks and societies. Parameters should be adjustable by the user and should not be embedded in code.

- *Streamline the processing tools so that they can be run on a daily basis*. Tools should be able to complete their processing overnight, and should ideally run in less than 30 minutes so that users can more easily experiment with different parameter values.

- *Demonstrate applications*. Applications of each process should be demonstrated (e.g., bus journey stages, full journeys, and the expanded OD matrix), from both the operational and passenger perspectives, and at various levels of aggregation.

## 1.3 THESIS ORGANIZATION

The methods developed in this thesis can be grouped into three categories: bus origin and destination inference, interchange inference, and full-journey matrix expansion. Since each category has a distinct methodological background, each is presented in its own chapter. Following an overview of London's public transport system and operations in Chapter 2, the three phases of the problem are presented in chapters 3, 4, and 5, with each including a review of the relevant literature and a discussion of their validation and results. The technical implementation of the methods is discussed in Chapter 6 followed by a demonstration of applications in Chapter 7. Chapter 8 reflects on the study's findings and presents its conclusions and recommendations.

# Public Transport in London

2

The methods presented in this thesis were developed and tested using data from London's public transport network. This chapter describes the network's structure and services, and introduces the data sources used to conduct this research.

## 2.1 London's public transport network

London's transit network is planned and managed—and partially operated—by Transport for London (TfL), a government body serving the Greater London metropolitan area and led by a 17-member board appointed and chaired by the Mayor of London (Transport for London 2011). TfL oversees most transportation services in Greater London, including public transport, taxis, traffic management (including congestion charging in the city center), river services, bicycle sharing, and paratransit.

### 2.1.1 *London Buses*

London Buses, a subgroup of TfL's Surface division, plans and manages the metropolitan area's network of over 800 routes and 21,000 stops, which provide roughly six million passenger trips daily. London's fleet of over 8,500 iconic red double-decker (and single-deck) buses is operated by private companies, who bid for contracts awarded by the agency (Transport for London 2011, 2012).

Bus riders are charged a flat fare, regardless of travel location or time of day (although some riders are eligible for discounts). Fares can be paid in cash on board (or compulsorily at ticket vending machines at some Central London bus stops) or riders can tap their Oyster cards upon boarding, to deduct credit or validate travel passes stored on the card.

### 2.1.2 *The London Underground*

The Underground, or *Tube*, is London's metro system, offering high frequency service on 11 lines, several of which include multiple branches (see figures 2-1 and 2-2).[1] Underground fares are assessed according to the geographic travel zone of the rider's starting and ending locations (Figure 2-1), requiring users to validate their fare media upon both entry and exit.

The Underground system is often at capacity during peak hours, and relieving this congestion is a major goal of many of TfL's fare policies and capital projects. Higher fares are charged during peak periods to incentivize off-peak travel, while recent and current projects such as the East London Line Extension and Crossrail were designed in part to provide alternatives to Underground service at key bottlenecks.

### 2.1.3 *National Rail*

National Rail is the brand name of the Association of Train Operating Companies (ATOC), a group of 24 private firms that provide commuter and intercity rail services throughout Great Britain. Several National Rail lines serve London, but all terminate outside (or slightly within) the bounds of the Circle Line, the quasi-orbital Underground line that loosely defines the city's center (see Figure 2-3). Much of the congestion on the Underground is due to passengers transferring between National Rail and the Tube during peak periods, and stations serving both Underground and National Rail lines exhibit some of the highest passenger volumes in London.

---

1 Although the term *Tube* originally referred to the seven deep-bore rail lines drilled into the earth (thus exhibiting the cylindrical tunnels and passageways that inspired the nickname), the term is now typically used to refer to both the deep-bore lines and the four "subsurface" lines, which were excavated directly from the streets above using the "cut-and-cover" technique.

### 2.1.4 *London Rail*

In addition to the Underground, TfL provides rail service on the London Overground, Docklands Light Railway (DLR), and Tramlink. All three modes are planned and managed by TfL's London Rail group, which also liaises with ATOC to coordinate National Rail services within Greater London (Transport for London 2011).

The Overground system comprises a series of rail lines primarily in travel zones two through four, which are being developed into a suburban orbital network encircling most of the Underground system. Some tracks are shared with National Rail and Freight services, and many stations were acquired from National Rail.[2]

The Docklands Light Railway is an automated system operating on dedicated rights of way in the redeveloped Docklands area, one of Greater London's three primary commercial cores. DLR services connect with Underground and Overground lines in East London, and one branch extends into another commercial core in the City of London.

The Tramlink system consists of three light rail lines serving Croydon, the southernmost borough in Greater London. Tramlink connects with Underground and National Rail services and, with the opening of the extended East London Line, the Overground network.

## 2.2 DATA SOURCES

### 2.2.1 *Oyster*

The Oyster card is accepted on most TfL transport modes and typically registers 16 million transactions daily, or roughly 10 million passenger trips. Oyster use is incentivized by reduced fares, designed to increase passenger throughput by streamlining fare validation at station gates and bus fareboxes.

---

2 A notable exception is the East London Line, a former Underground service that was extended along a combination of disused above-grade rail viaducts and newly acquired rights of way (Ng 2011, Muhs 2012).

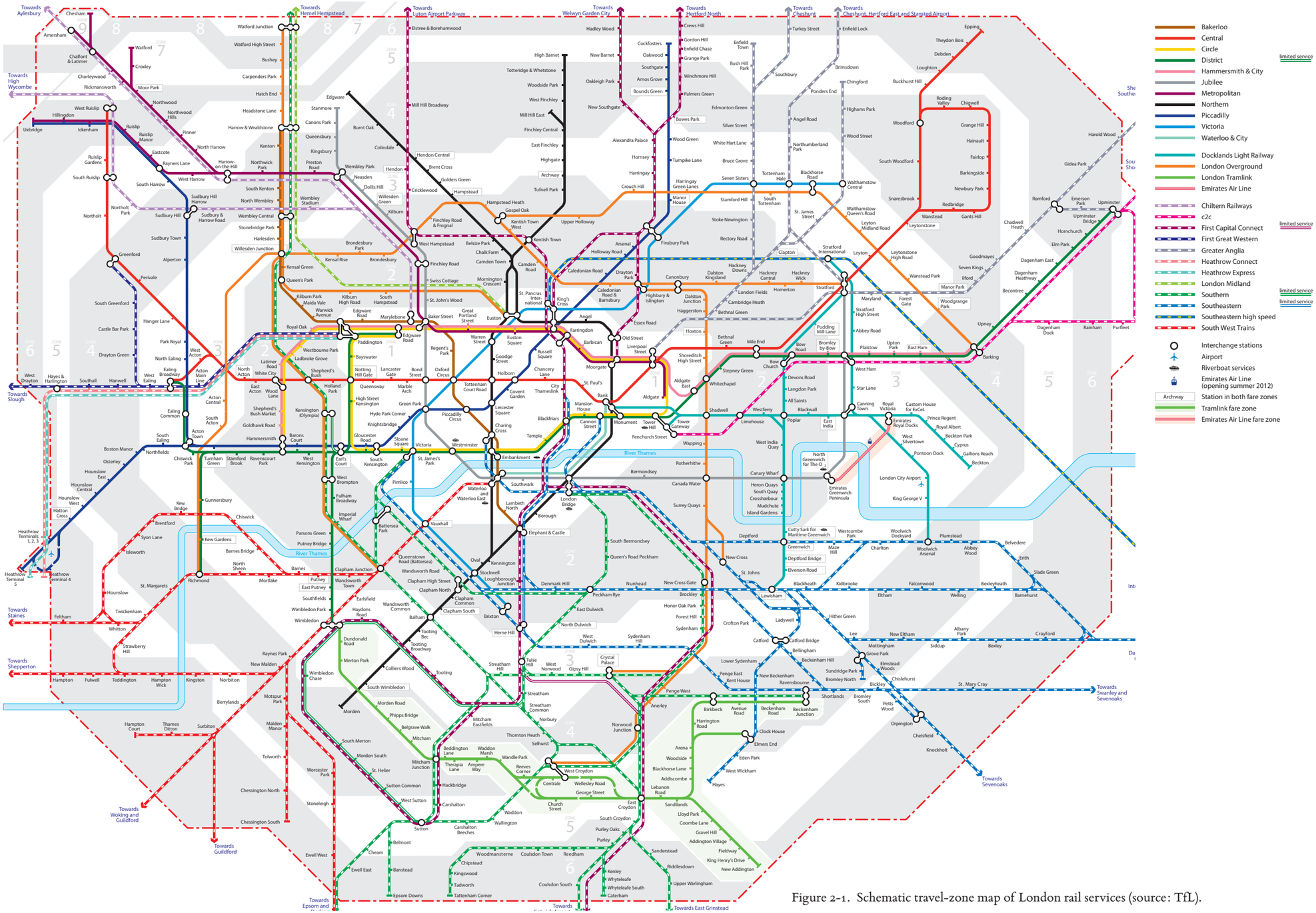


Figure 2-1. Schematic travel-zone map of London rail services (source: TfL).

Figure 2-2. Geographic map of London rail services (source TfL).

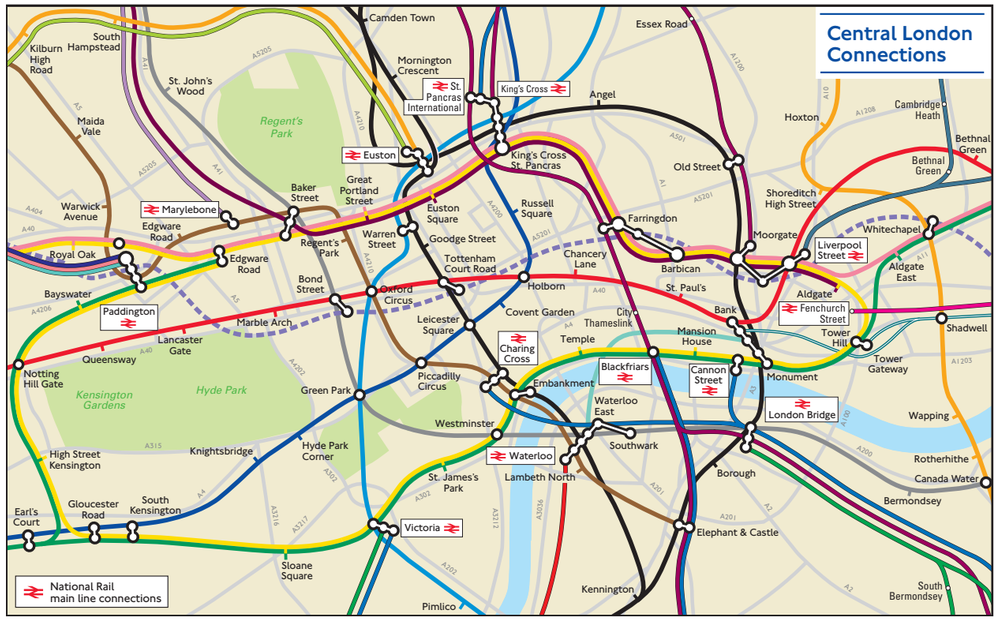


Figure 2-3. Underground, National Rail, and DLR services in Central London (see Figure 2-2 for legend; source: TfL).

Oyster cards can store prepaid weekly, monthly, or annual *Travelcards* valid for unlimited use on various modes within specified travel zones, and can also store monetary credit, which can alternatively be used to pay for each trip individually (referred to as *pay as you go*). All Oyster users are required to tap their cards when boarding buses or when entering or exiting station gates, but some stations are ungated. Travelcard holders are not required to tap at ungated stations,[3] but Oyster readers are provided for pay-as-you-go users, who are required to tap their cards in order to deduct the correct fare.

Oyster data is retained by TfL for eight weeks before being archived[4] and the complete population of Oyster data for one week of every month

3 Travel beyond the zones (or time periods) covered by a Travelcard requires the customer to pay an additional fare (the difference in fare between travel covered by the card and the trip taken by the customer) using pay as you go.

4 For privacy, identifying information is removed prior to archival.

was made available for this research. Card identifiers are encrypted for privacy, and data are exported from a TfL database in plain-text format.

Oyster entry and exit data are available for Underground, Overground, and DLR transactions at gated stations (and at ungated stations for pay-as-you-go transactions), while boarding (but not alighting) information is available for buses and trams.[5] The Oyster card was recently made available for National Rail travel, and its use on the mode has been steadily increasing (Muhs 2012).

### 2.2.2 *iBus*

iBus is TfL's AVL system, and is installed on the entire fleet of buses. While several data sets are recorded by the system, this research uses a set containing a record for each *stop event*: an instance of a vehicle serving (or passing) a bus stop (TfL 2006, Hardy 2007, Robinson 2010, Continental Automotive 2011, Hounsell et al 2012).

iBus tracks each vehicle's location using a combination of GIS, tachometer, speedometer and gyroscope data. The system attempts to record information about the time at which each bus neared a stop, opened its doors, closed its doors, and pulled away from the stop. If all four data are recorded internally, it reports the door opening time as the stop arrival time and the door closing time as the stop departure time. If either door event is unavailable, the approaching or departing timestamp is used. If only one of the four events are recorded, the same time is written to both the arrival and departure fields.

iBus typically records approximately 5 million transactions daily. Data are provided to MIT researchers through a secure database reporting interface, so that data can be obtained for the same periods for which Oyster data are collected.

### 2.2.3 *Gateline and ETM*

Aggregate counts of passenger activity are provided by rail station gates, or *gatelines*, and at bus fareboxes, or *electronic ticket machines* (ETMS). Rail

5 Modes requiring both entry and exit taps store a record for each. Entry records contain entry information (as expected), while exit records store both entry and exit information in order to assess the correct fare.

counts are aggregated by date, hour, station, direction (entry versus exit), and ticket type. ETM data are similarly aggregated by date, hour, and ticket type, plus the bus's direction (inbound versus outbound) and its route. ETM data were available for all bus routes, but gateline data for several gated National Rail and Overground stations were unavailable (although these data are recorded and may be available for future analyses).

### 2.2.4 *Spatial Data*

In addition to Oyster, iBus, gateline, and ETM data, some of the methods presented in this thesis require spatial information. A list of bus stops was acquired from TfL, which includes spatial coordinates. A similar table of rail stations was also acquired, although it lacked spatial information. Using each station's unique identifier, the data were joined to a similarly identified GIS layer to obtain spatial coordinates.

# Bus Origin and Destination Inference

3

In order to observe entire passenger journeys using fare-transaction data, time and location information must be obtained or inferred about the start and end of each journey stage.[1] The zonal fare-pricing structure of most rail modes in Greater London necessitates fare transactions at both the start and end of each stage, yielding time and location data for both the entry and exit stations.[2] But London Buses, like many other urban bus systems, charges flat rather than zonal fares, enabling fare transactions to be completed at the time of boarding and therefore requiring no alighting transaction.

Oyster bus records contain information about the time of the transaction and about the vehicle, route, and vehicle-trip number, but information about the alighting location and time must be inferred. Furthermore, the onboard AFC equipment does not obtain location data from the bus's

1 The concept of a journey stage is defined in Chapter 4, where it is relevant to the inference of interchanges. In the context of bus travel, however, a bus journey stage can be considered one cardholder's ride on a single vehicle, regardless of whether that person made another bus (or rail) trip immediately before or after it.

2 As discussed in Chapter 2, all London Underground, most London Overground, and many National Rail stations are gated, requiring transactions for both entries and exits. Ungated stations, including the remaining Overground and National Rail stations, and most DLR platforms, require Oyster users to tap their cards only if they are paying a fare using stored credit. Customers with travel passes valid for the journey stage are not required to tap their cards at either end.

AVL system.[3] The boarding location must therefore also be inferred, by merging the two data sets offline.

This chapter describes the process by which boarding and alighting locations and alighting times are automatically inferred for the population of Oyster bus transactions for a given day. The resultant data are a required input to the interchange-inference process, which in turn is an input to the full-journey matrix estimation process (both of which are described in subsequent chapters). In addition to their applicability to full-journey analysis, OD-inferred bus data are useful for a number of bus-only other applications, which are presented in Chapter 7.

## 3.1 PREVIOUS RESEARCH

### 3.1.1 *Inferring Bus Origins Using AFC and AVL Data*

On bus systems for which AFC and AVL data are available, passenger boarding locations are typically inferred by matching each AFC record to an AVL record, then using the AVL location as the passenger's boarding location. In most cases the records can be matched by first selecting the subset of AVL records that share the same vehicle, route, or vehicle trip as the AFC record, and then by selecting from that subset an AVL record that has a timestamp closely matching that of the fare transaction.

Zhao et al (2007) propose this approach in their study of the Chicago Transit Authority's bus and rail system, and Cui (2006) applies the methodology to the city's bus network. Chicago's AFC data uniquely identify specific vehicles, and each AVL record indicates the time at which a bus opened its doors. By including only those AFC transactions that occur within five minutes after an AVL event, Cui infers the origins of approximately 90 percent of all bus boardings.

Wang et al (2011) apply a similar approach to five TfL bus routes, matching AFC transactions to the (temporally) closest AVL record, regard-

3 TfL's new fare-payment system, currently in development, will record location data with each fare-transaction record.

less of whether it occurs before or after the fare transaction.[4] Following Cui, Wang limits matching to within a five minute range (in this case either before or after the time of the AFC transaction), yielding a similar origin-inference rate of 90 percent. Munizaga et al (2011) infer origin locations for 98.5 to 99.9 percent of all bus boardings on Chile's Transantiago network, but it is unclear whether a time limit is being applied.

### 3.1.2 *Inferring Bus Destinations Using AFC and AVL Data*

The absence of AFC alighting transactions is common to many bus systems. Lacking this information, previous studies have inferred alighting locations under the assumption that the most likely place for a passenger to alight a bus is the stop closest to her next bus boarding (or station entry) location. Barry et al (2002) apply this assumption in their study of New York's subway system, and later on the city's combined bus and subway network (2009), both of which record fares only upon entry. By making the additional assumption that a rider's initial origin of the day is a close approximation of her final daily destination, they are able to infer destinations for passenger journey stages regardless of a stage's sequence within a rider's daily travel history.

Barry et al (2002) tested these assumptions against a travel-diary survey, showing that the assumptions were valid in 90 percent of the cases observed. Additionally, Navick and Furth (2002) applied the same assumptions in their study of five Los Angeles bus routes, validating the assumptions against ride-check data which showed that the assumptions were valid on four of the five routes studied.[5]

The two aforementioned destination-inference assumptions have since been applied in several other studies of transit networks with entry-only AFC data. Zhao et al's (2007) work on the CTA rail system inferred destinations for 71 percent of AFC passenger trips, inferring destinations only when the candidate alighting location is within a 400-meter Euclid-

4 Multiple event types can trigger the recording of an AVL event in TfL's system, which can sometimes lead to ambiguity about whether a bus was arriving at or departing from a stop (see section 2.2.2 for details).

5 Navick and Furth infer the validity of these assumptions by testing the symmetry of each route's boarding profile—the pattern of daily boardings in one direction should match the pattern of daily alightings in the other. They hypothesize that the lack of symmetry on the one route that failed their test was due to its sharing of a corridor with other routes.

ean distance of the rider's next origin. Cui (2006) applied a 1,110-meter rectilinear distance limit to the CTA's bus network, achieving a destination-inference rate of 67 percent. Trépanier et al (2007) applied similar logic to the bus network of Gatineau, Québec, using a two-kilometer Euclidean distance limit and inferring destinations for 66 percent of the observed transactions.

Wang et al (2011) apply the closest-stop and last-of-day assumptions in their London study, in which AVL data are used to infer passenger alighting times after the inference of alighting locations. As in the origin-inference process, AVL and AFC data are matched by route and vehicle trip, but the AVL record is chosen from the resultant subset by matching its bus stop code to that identified using the two destination-inference assumptions. Studying five routes and imposing a one-kilometer distance limit around each route, Wang et al infer destinations for 57 percent of the observed AFC data. Finally, the inference process is validated against TfL's Bus Origin–Destination Survey (BODS), which exhibits a similar distribution of alighting locations for the observed routes.

While most previous studies inferred destinations by using SQL queries that required manually generated lookup tables or spatial buffers, Munizaga et al (2011) automate the process by integrating a database with custom software.[6] To account for bus-route circuity and for the absence of trip direction information in Transantiago's AVL and AFC data, the closest-stop rule was modified to minimize a generalized time value, which accounts for both the distance between stops as well as the time savings realized by alighting at an earlier stop and walking. With roughly six million fare transactions per weekday on the combined bus and metro network, the process took several hours to complete but inferred approximately 83 percent of bus destinations.

## 3.2 ORIGIN INFERENCE

The origin-inference process builds upon previous research by incorporating and refining some of the aforementioned assumptions while taking advantage of additional features in the iBus data set. Performance is ad-

6 Barry et al (2009) automate an OD-inference process which does not use AVL data.

dressed by implementing the process entirely in an object-oriented programming language and tuning the code for speed and memory efficiency. To enhance compatibility with databases and analysis tools, data are input and output in an easily parsable comma-separated text format.

Although the three inference processes are conceptually performed in series, they are combined in a single application which performs some parts of each in parallel for efficiency. For clarity, however, the origin-inference algorithm is treated in this section as a discrete process.

### 3.2.1 *Exploratory Analysis of Input Data*

Following Zhao et al (2007), Cui (2006), Wang et al (2011), and Munizaga et al (2011), this methodology infers bus origins by matching AFC and AVL data. While Oyster and iBus are the only data sets used in the origin-inference process, additional data from TfL's Bus Contract Management System (BCMS) are used in this section to empirically demonstrate the relationship between the AFC and AVL data.

Wang (2010) notes that Oyster timestamps are truncated to the minute, and that their seconds component can be treated as a random variable uniformly distributed between 0 and 59, inclusive. By adding 30 seconds to each Oyster timestamp, an approximation of the expected time value can be obtained. The BCMS system records fare-transaction data in tandem with the Oyster system, but contains only a subset of Oyster's fields. The BCMS data do, however, retain the seconds component of the transaction time, and can be used to illustrate the error introduced by the truncation of Oyster timestamps.[7] Figure 3-1 shows the effect of this truncation, as all fare transactions are effectively shifted to the midpoint of the minute in which they occur.[8]

The figure also illustrates the times recorded by the iBus system, which usually include both arrival and departure timestamps (Wang et al had access only to departure times). In most cases BCMS shows fares being paid between an iBus event's arrival and departure times, and these

7 Although the matching of BCMS and Oyster data would improve the resolution of Oyster timestamps, the inefficiencies of obtaining and processing complete daily sets of BCMS data (in addition to the large Oyster and iBus data sets) do not justify this relatively minor improvement.

8 Data displayed for a single bus serving a portion of route 38 eastbound, on Saturday, 26 February 2011

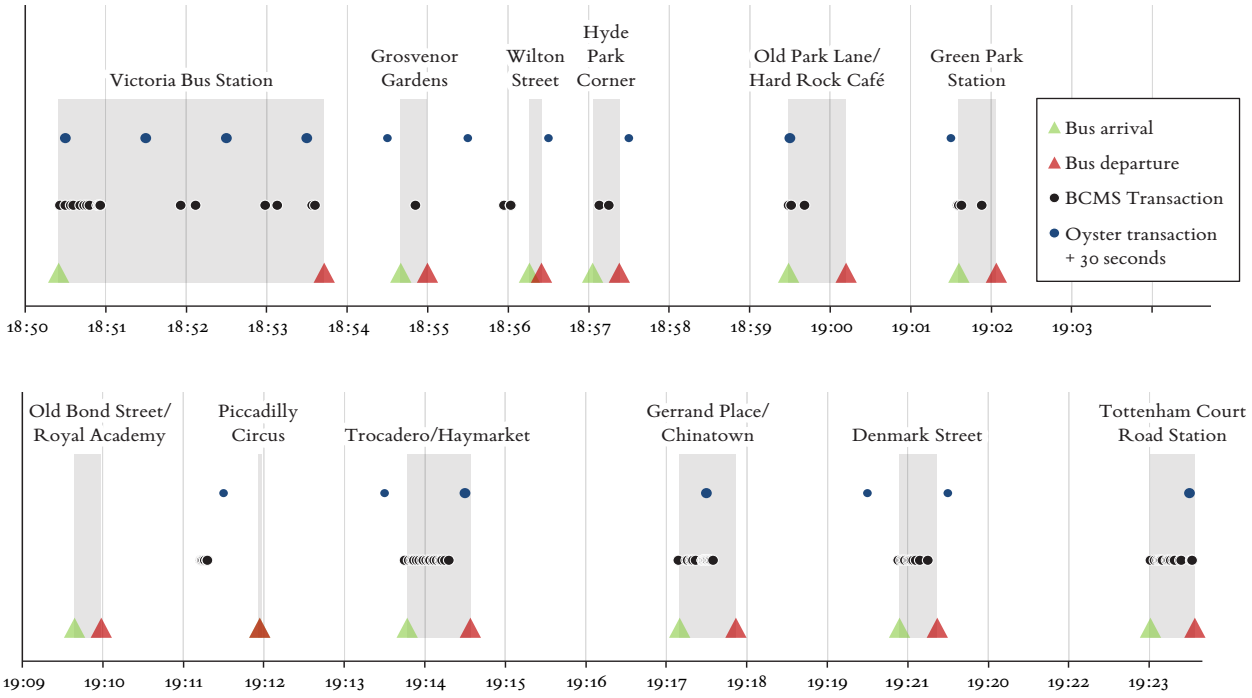


Figure 3-1. Oyster, BCMS, and iBus data for a portion of a single bus (route 38 eastbound).

transactions are often concentrated near the arrival. This coincides with the expected behavior of customers waiting at a bus stop and then paying their fares in succession after the door opens. Some of these clusters of transactions begin a second or two before the arrival event (e.g., Trocadero, Gerard Place), possibly due to a difference between the Oyster reader's internal clock and that of the iBus system, which is synchronized to GPS satellites. These small differences are unlikely to have any effect on the origin-inference process, but two of the stops shown have a more significant offset.

Two BCMS transactions occur approximately 40 seconds before the iBus record at the Piccadilly Circus stop, which contains the same time for the bus's arrival and departure. As noted in section 2.2.2, iBus will use the same time as a stop's arrival and departure if one or the other are missing or if the bus passed the stop without opening its doors (Robinson 2007). It is therefore unclear whether this iBus record marks the time at which the bus drew near to the stop, opened or closed its doors, or departed. Since separate arrival and departure times were recorded for the previous stop, it is perhaps most likely that the Piccadilly event represents the door clos-

ing or departure time, and that the customers in question boarded at that stop after an unrecorded arrival or door-opening event.

The other anomaly in the example is that two customers appear to tap their cards between the Grosvenor Gardens and Wilton Street stops, approximately 20 seconds before the latter. Since distinct arrival and departure times are recorded for both stops, iBus is assumed to have recorded—at a minimum—the times during which the doors were open, or an even earlier arrival (or later departure) time if the system used GPS or odometer data in the absence of door events. In either case, the AVL system indicates that the customers tapped their cards between the two stops. It then follows that either the customers boarded at Grosvenor Gardens (or earlier) and did not tap their cards until after the bus was in transit, or that there was an unexplained error in the recording of one or more iBus events or both Oyster events.

The first step in Wang's (2010) origin-inference method, in which Oyster and iBus records are matched if they occur during the same minute, is accurate only if no more than one iBus event occurs during that minute. This criterion is satisfied in the cases shown in Figure 3-1, but the Wilton Street and Hyde Park Corner stop events are very close to sharing the same minute, and it should be expected that multiple stops would occur during the same minute when viewing a large sample of iBus data. That condition is therefore not applied in this methodology: instead, all available data are matched using Wang's second criterion, in which Oyster transactions are assigned to the iBus event closest in time.

But an additional test can be applied before testing for temporal proximity. The AVL data used in Cui's (2006) study contained door opening times while the data used by Wang contained only departure times, although iBus departure times can in some cases be ambiguous. Since the iBus data used in this study contain both arrival and departure times (notwithstanding the occasional ambiguities), Oyster transactions can in many cases be observed to fall between an iBus record's arrival and departure times, obviating the need for further temporal tests.

Almost all BCMS records in Figure 3-1 fall between iBus arrival and departure times, but a degree of error is introduced by the use of Oyster data. The fare transactions shown by BCMS to have occurred while the bus was stopped at Victoria, Old Park Lane, Gerrard Place, and Tottenham Court Road still occur during these stops when viewed through their corresponding Oyster records. But other transactions recorded within

stops by BCMS are shifted between stops, such as the taps at Grosvenor Gardens or Hyde Park Corner.

For the Oyster transactions that do not occur within an iBus event, applying Wang's rule of matching Oyster data to the temporally closest iBus event would result in almost the same result as if the Oyster data contained the BCMS timestamps. The boardings at Green Park, for example, would be shifted from within their stop event to a time before it, but would still be matched to the same event due to their temporal proximity. In the example, the sole difference between the Oyster and BCMS approaches occurs between Grosvenor Gardens and Wilton Street. The two fare transactions, shown by BCMS to occur only a few seconds apart, occur during two different minutes. The earlier transaction is shifted back roughly 30 seconds, assigning it to Grosvenor as the closest stop rather than Wilton, while the later transaction is shifted past Wilton, which continues to be its closest match.

Based on the example in Figure 3-1, it appears reasonable to assume that the correct stop can often be inferred using Oyster and iBus data, but that in some cases the previous or next stop will be chosen instead.

### 3.2.2 *Origin-Inference Methodology*

Central to the origin-inference process is the matching of AFC and AVL data. Since there is a many-to-one relationship between fare transactions and stop events (each boarding is associated with a single stop event but each stop event can be associated with multiple passenger boardings), the relationship can be specified by the fare transaction. The following methodology is therefore performed once for each Oyster record.

The origin-inference process is described in Figure 3-2. If an Oyster record represents a bus transaction, and if iBus data exist for the specified route and trip number, 30 seconds are added to the Oyster start time, which is compared to the list of stops for the specified route and vehicle trip. If the offset Oyster time falls between a stop event's arrival and departure time (as observed at Gerrard Place in Figure 3-1), that stop event is selected. If the Oyster transaction occurred between two stops, it is assigned to the closer of the previous stop's departure time and the next stop's arrival time. If the Oyster transaction occurs before the first stop, the first stop is chosen. If the Oyster transaction occurs after the final stop, or if the final stop was chosen because the Oyster transaction preceded it

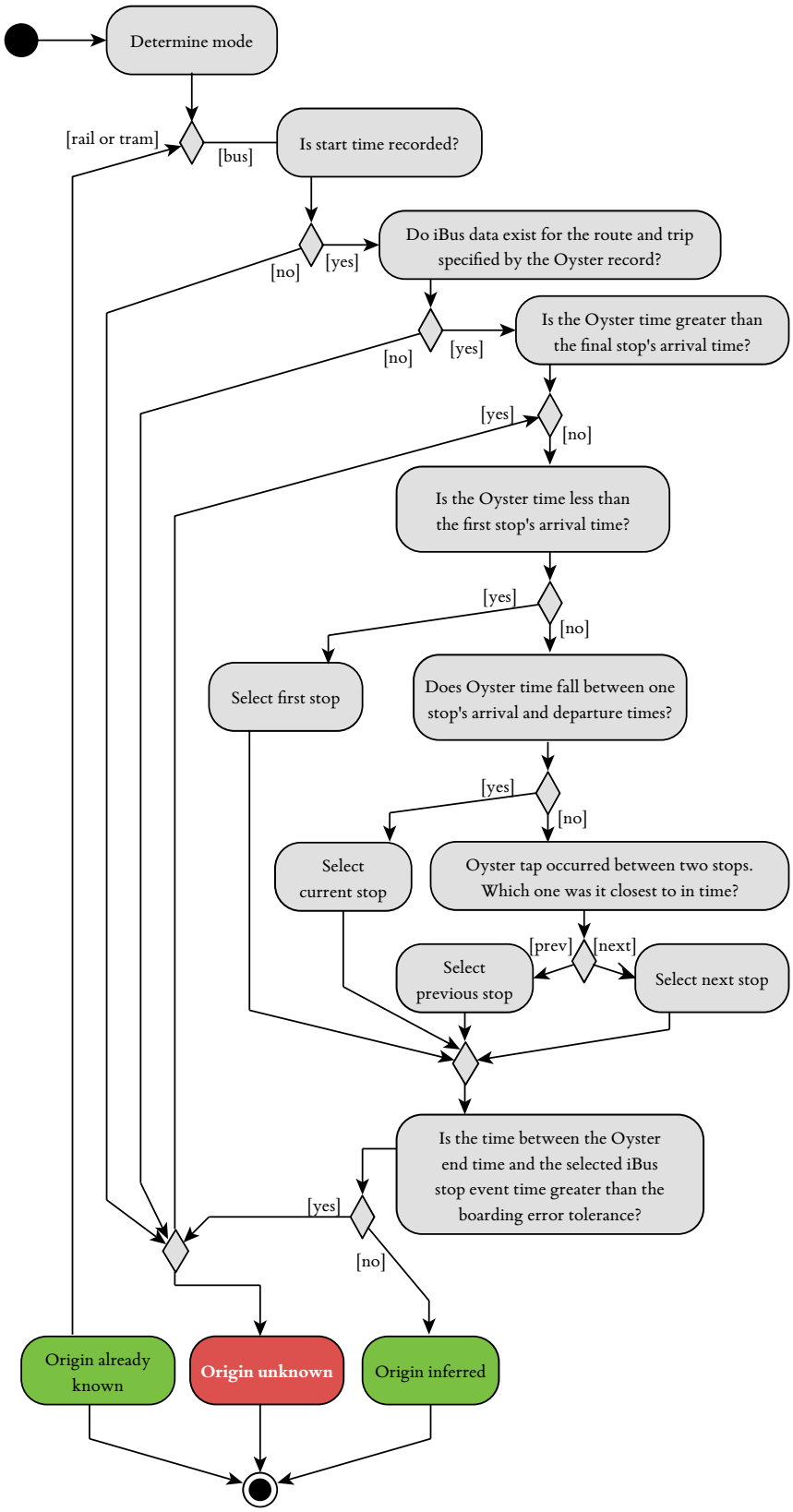


Figure 3-2. Activity diagram of the origin-inference process.

and was closest to it, no boarding location is inferred because it is assumed that riders do not board at the final stop of a vehicle trip.

If a stop event was chosen during the previous step, an *origin-inference error* is calculated. This value is defined as the difference between the offset Oyster time and the closest iBus time: the arrival time if the next stop is chosen, the departure time if the previous stop is chosen, or zero if the Oyster transaction falls within a stop event. The origin-inference error is appended to the Oyster record, and if the magnitude of the error is greater than the user-defined maximum, the boarding location is considered to be unknown. Otherwise, the bus stop ID of the chosen stop event is appended to the Oyster record as the inferred boarding location.

### 3.2.3 *Origin-Inference Results and Sensitivity Analysis*

The origin-inference processes was implemented as a Java application (discussed in Chapter 6) and was tested using complete daily sets of Oyster and iBus data for all of Greater London over ten consecutive weekdays (the 6th through 10th and 13th through 17th of June, 2011). Each day's data consisted of 6.1 to 6.5 million bus transactions, with a daily average of 6,295,634. The software performs the origin-, destination-, and interchange-inference processes on one day's worth of data in less than 30 minutes, although the origin-inference portion is typically finished within the first five minutes.[9] The program keeps a running total of origin-inference statistics, which are written to a report after all Oyster records have been processed. The contents of this report are shown in Figure 3-3 and in tables Table 3-1 and Table 3-2

Table 3-1 shows the rates at which the various origin-inference rules were used to match or discard Oyster data.[10] Less than two percent of transactions were discarded because of incomplete or missing data, while nearly 28 percent of all Oyster bus transactions, after timestamp truncation and the 30-second offset, fell between a bus's arrival and departure time.

---

9 The software was run on a 2.8 gigahertz Intel Core i7 machine with 8 gigabytes of RAM running the Windows 7 operating system.

10 The statistics in Table 3-1 were averaged over the ten-weekday period, and were measured before the application of the maximum origin-inference error parameter.

Table 3-1. Detailed origin-inference results, ten-day average.

| Result | Count | Percentage |
|---|---|---|
| Cannot infer because bus route data not included in Oyster record | 36,210 | 0.57% |
| Cannot find route in iBus data | 24,490 | 0.39% |
| Cannot find trip in iBus data | 61,123 | 0.96% |
| Within stop event | 1,764,673 | 27.82% |
| Between stop events; closer to previous | 1,632,375 | 25.73% |
| Between stop events; closer to next | 2,238,488 | 35.29% |
| Before first stop event | 487,865 | 7.69% |
| After last stop event | 98,561 | 1.55% |
| Total | 6,343,784 | |

Figure 3-3 shows the distribution of origin-inference error for the ten-weekday period (the daily average distribution can be observed by simply dividing the frequencies by ten).[11] The distribution is discontinuous at zero because it is piecewise defined: as discussed in section 3.2.2, the negative portion of the distribution is calculated using iBus arrival times, the positive portion is calculated using departure times, and the frequency of 18.4 million at zero accounts for the Oyster transactions that occurred between iBus arrival and departure times.

Beyond its discontinuity, the origin-inference error histogram exhibits two other notable traits. First, the distribution is visibly skewed to the left. Since negative error values indicate Oyster transactions that occurred before iBus arrival times, this rise in the left tail fits the expected result of more Oyster transactions occurring near a bus's arrival time than its departure time, as illustrated with the Victoria stop in Figure 3-1. Second, the distribution rises sharply near -30 seconds and falls sharply again at 30. This resembles the expected result of adding the aforementioned skewed, discontinuous distribution to a uniformly distributed function on the interval [-30, 30]. The uniformly distributed component of the curve in

11 Negative errors in the histogram denote Oyster transactions that occurred before the closest bus arrival, while positive errors indicate transactions that occurred after the closest bus departure. Oyster transactions that occurred between a bus's arrival and departure times are assigned an error of zero.

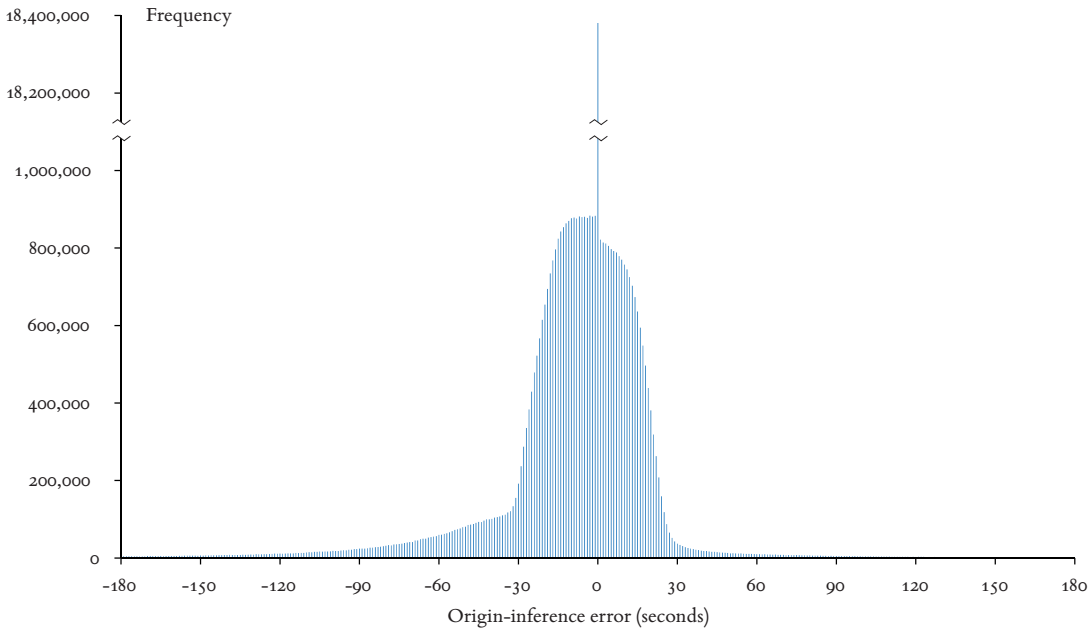


Figure 3-3. Histogram of origin-inference error for all Oyster bus boardings

turn matches an expected side effect of using truncated and offset Oyster time values rather than BCMS values. If a rider taps his card between a bus's arrival and departure time, but if that stop event does not span the midpoint of a minute, the 30-second offset of the minute-resolution Oyster timestamp will occur up to 30 seconds before the arrival time or up to 30 seconds after the departure time. For example, Figure 3-5 shows that a rider tapped after the bus's arrival at Grosvenor Gardens but before its departure. Since the stop event did not span the midpoint of the minute (18:54:30), the offset Oyster record appears a few seconds before the arrival time, introducing a small error.

In rare cases, bus route or trip data are recorded incorrectly, causing some Oyster records to be erroneously matched to iBus records several hours away. This noise adds long tails to the origin-error distribution, resulting in 96.0 percent of origin-inference error values falling within a tolerance of ±2 minutes. Increasing this threshold to ±5 minutes raises the proportion by only 1.4 percent, illustrating that this choice of thresholds sits well beyond the sensitive portion of the parameter's range. While the choice of this five-minute margin can be seen as arbitrary, it seems reasonable to expect an occasional five minute difference between AFC and AVL times, especially since buses sometimes stop for several minutes at busy

stations or signals during peak hours, and because customers might in some cases tap their cards after the bus has left the stop. The five-minute boarding-error tolerance (also used by Cui [2006] and Wang et al [2011]) is therefore applied in this study. This value, however, can be easily changed by the user at runtime.

The results of the origin-inference process on the ten-day sample, after the application of the five-minute error threshold, are summarized in Table 3-2. The process was unable to infer boardings for 1.36 percent of bus transactions, while an additional 2.61 percent were matched but excluded because their origin-inference errors exceeded the five-minute limit. The origins of the remaining 96 percent of Oyster bus trips are inferred, for an average of approximately six million bus stages per day.

| Result | Count | Percentage |
|---|---|---|
| Not inferring bus boarding because Oyster record was not matched to an iBus record | 85,441 | 1.36% |
| Not inferring bus boarding because the time between its boarding tap and the nearest iBus event is beyond ±5 minutes | 164,566 | 2.61% |
| Bus stages with origin inferred within ±5 minutes | 6,045,627 | 96.03% |
| Total | 6,295,634 | |

## 3.3 DESTINATION INFERENCE

Like the origin-inference process, the process for inferring bus alighting times and locations builds upon previous research to create a more robust and efficient algorithm. As stated in the previous section, the three sequential processes are performed partly in parallel for efficiency, but the destination portion is presented here as a logically discrete process for the sake of clarity.

### 3.3.1 *Input Data*

The primary input to the destination-inference process is the origin-inferred Oyster data, to which the rider's inferred destination will be appended. The process is founded on the two key assumptions introduced by Barry et al (2002): that the best estimate of a rider's alighting location is the stop closest to that rider's next origin, and that the best estimate of a rider's final daily destination is that rider's first daily origin. The destination-inference process therefore requires spatial information about the rider's possible alighting locations and about her subsequent origin. In the case of London, this information is contained in two data sets: one of bus stops and another of rail stations, including Underground, Overground, National Rail, DLR, and Tramlink modes. The fourth input to the process is iBus data, which is used to infer alighting times as demonstrated by Wang et al (2011).[12]

The only information required of stations are the unique identifiers by which they are referenced in the Oyster data—which in this case is the British National Location Code (NLC)—and the station's Cartesian coordinates. Station coordinates were copied from GIS data obtained from TfL but were then converted from geographic coordinates (measured in angular degrees) to linear coordinates (measured in meters) using the British National Grid projected coordinate system. The benefits of this conversion are, first, that it avoids the spatial distortion that results from the convergence of meridians at non-equatorial latitudes (such as London's) and, second, that meters provide a satisfactory resolution for pedestrian-scale transportation analysis (which is required for the inference of alightings and interchanges). By contrast, the use of degrees of latitude and longitude would require the processing of real numbers rather than integers, which would double or quadruple the amount of hardware resources required to process spatial data[13].

Bus stop data similarly require only a unique identifier and spatial coordinates. The "stop code" field from the BusNet system is used, since it is guaranteed to be unique and to persist indefinitely. iBus data, which

12 See section 3.1.2.

13 The software stores metric coordinates and distances using Java's `short` and `int` data types, which comprise 16 and 32 bits, respectively. Real numbers would be processed using the 64-bit `double` data type.

use their own internal, temporary bus stop identifiers, are assigned the appropriate BusNet stop codes when exported from TfL's servers, enabling Oyster and iBus data to be matched in the software. Each stop's spatial coordinates are stored by BusNet in the British National Grid coordinate system, which is why that system in particular (among many projected metric coordinate systems) was chosen for the station data.

Data were obtained for 21,554 bus stops, which describe the locations of each bus stop's pole marker. In many cases, multiple stops are clustered near a single logical node on the transport network, such as an intersection or rail station (Figure 3-4 and Figure 3-5). Data were also obtained for

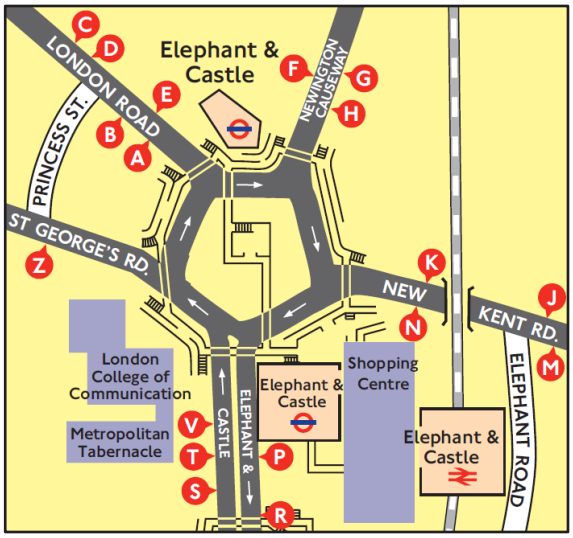


Figure 3-4. Letter-designated bus stops near Elephant and Castle Tube station (source: TfL).

Figure 3-5. Elephant and Castle stops S and T, each having its own pole marker and shelter.

1,435 stations of various rail modes, 1,361 of which were spatially unique (in some cases multiple codes relate to different modes or gate clusters at the same physical station). These stops and stations are mapped in Figure 3-6, which encompasses all of Greater London. Most of the stations beyond the extent of the bus network belong to the National Rail system, which extends through all of Great Britain.

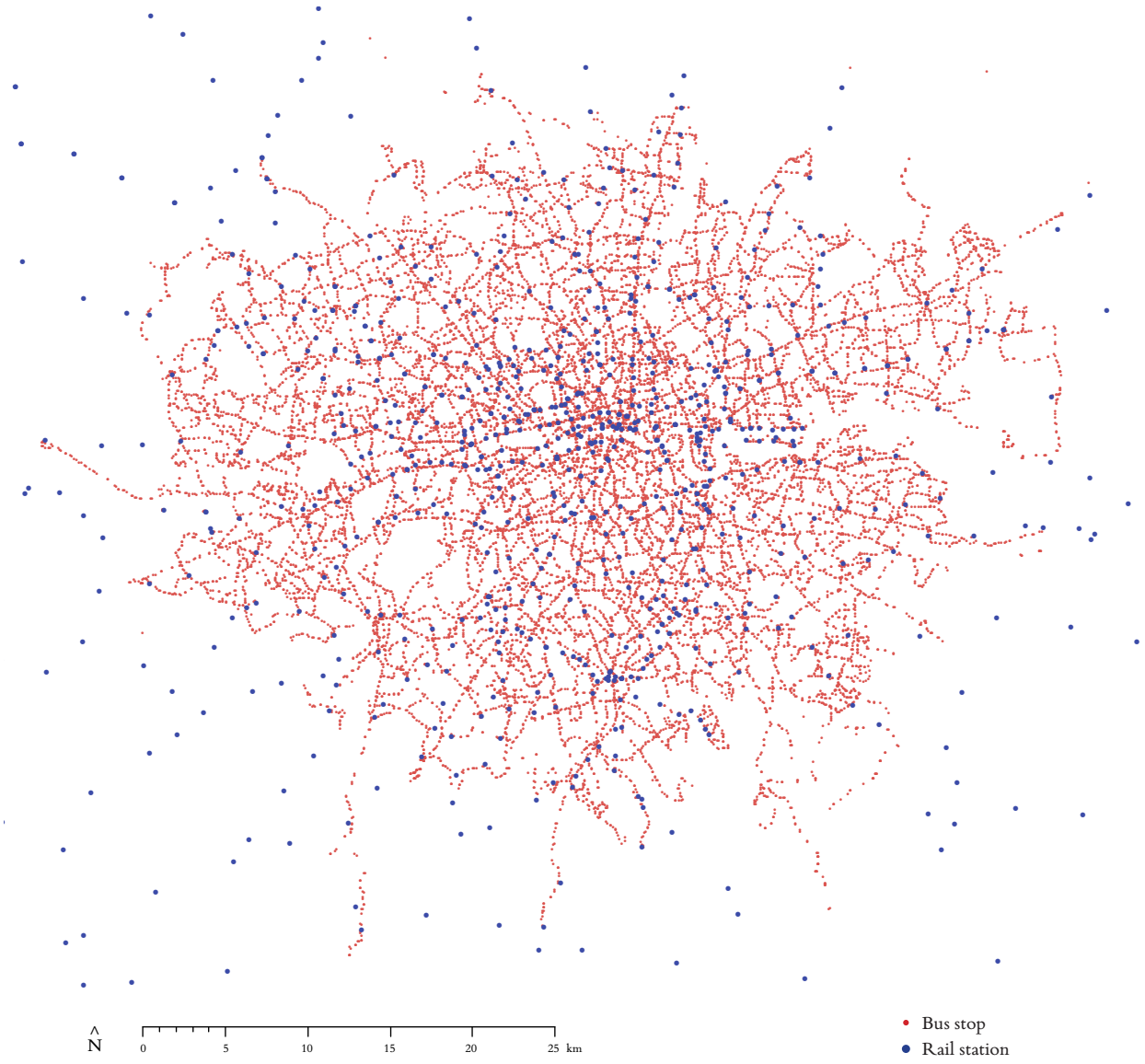


Figure 3-6. Stop and station locations in Greater London.

### 3.3.2 *Destination-Inference Methodology*

The destination-inference process draws from several of the methodologies described in section 3.1.2, using a refined processing algorithm to distill the destinations of over six million passenger bus trips from roughly 16 million Oyster transactions and five million iBus records in less than ten minutes.

The logic of the destination-inference process is built upon Barry et al's (2002) two assumptions: that riders alight at the stop closest to their subsequent transaction location or, in the case of the last trip of the day, that they alight at the stop closest to their first daily origin. These assumptions will be referred to herein as the *closest-stop rule* and the *daily-symmetry rule*, respectively.

It should be apparent to anyone who has ridden an urban bus, however, that there are many cases in which these two assumptions are not valid. The closest-stop rule, for example, is regularly violated by any rider who alights a bus, walks alongside the bus route to visit two or more businesses, then makes his next fare transaction at a stop or station that is closer to some other stop on the previous route than it is to the stop at which he actually alighted. The daily-symmetry rule is similarly violated by any rider whose final Oyster trip of the day ends somewhere other than his first Oyster origin, whether because he started and ended his day in different places, or because his first trip was preceded (or his last trip was followed) by a non-Oyster trip, such as a bicycle or taxi ride. Despite these shortcomings, the closest-stop and daily-symmetry rules have been shown to provide a reasonable approximation of riders' alighting locations, as discussed in section 3.1.2. In the absence of observed alighting information, Barry et al's assumptions reflect our best estimates of where riders might have ended their bus trips.

The closest-stop and daily-symmetry rules necessitate the calculation of several distances. The determinant location—the origin of the subsequent trip or the first origin of the day—shall be referred to as the *target location*. The distance between the rider's target location and each stop served by the current vehicle trip's *pattern* (a distinct sequence of bus stops served by one or more vehicle trips on a bus route) must therefore be calculated in order to determine which stop is closest. To accommodate these calculations, the set of all stops within each pattern, and the set of

all patterns within each route, are distilled from the daily set of observed iBus data before any destinations are inferred.

Once spatial information is known for all bus patterns, the algorithm illustrated in Figure 3-7 is applied to each Oyster record. First, duplicate Oyster records are discarded, since riders might sometimes tap a card in error before making a successful transaction. Valid non-duplicate transactions are then processed according to their mode: rail destinations should already be known, tram destinations are not inferred because AVL data are unavailable, and bus transaction are passed to the next step in the process so that their destinations might be inferred.

If a bus transaction is the only record associated with the card that day, it is assumed that any other travel that day was made without the Oyster card—possibly on a non-Oyster or private transport mode. The destination of the journey stage is therefore unknown. If there was more than one transaction made with the card that day, however, the current stage's origin-inference error is tested. If the magnitude of the error is greater than the maximum origin-inference error (specified by the user), the origin is considered to have been unreliably inferred, and it is assumed that the destination cannot be reliably inferred either.

If the bus stage's origin has been satisfactorily inferred, the target location for the destination-inference process is defined. If the journey stage is the cardholder's last that day, the origin location of his first trip is selected as the target location. Otherwise, the origin of the cardholder's subsequent trip is selected.

If the target location is unknown, or is a bus boarding that was not inferred within the maximum origin-inference error, the destination of the current trip is considered unknown. Otherwise, the algorithm attempts to find spatial coordinates for the current bus pattern and for the target location. If the coordinates of either the target node or the pattern are missing, an alighting location is not inferred.

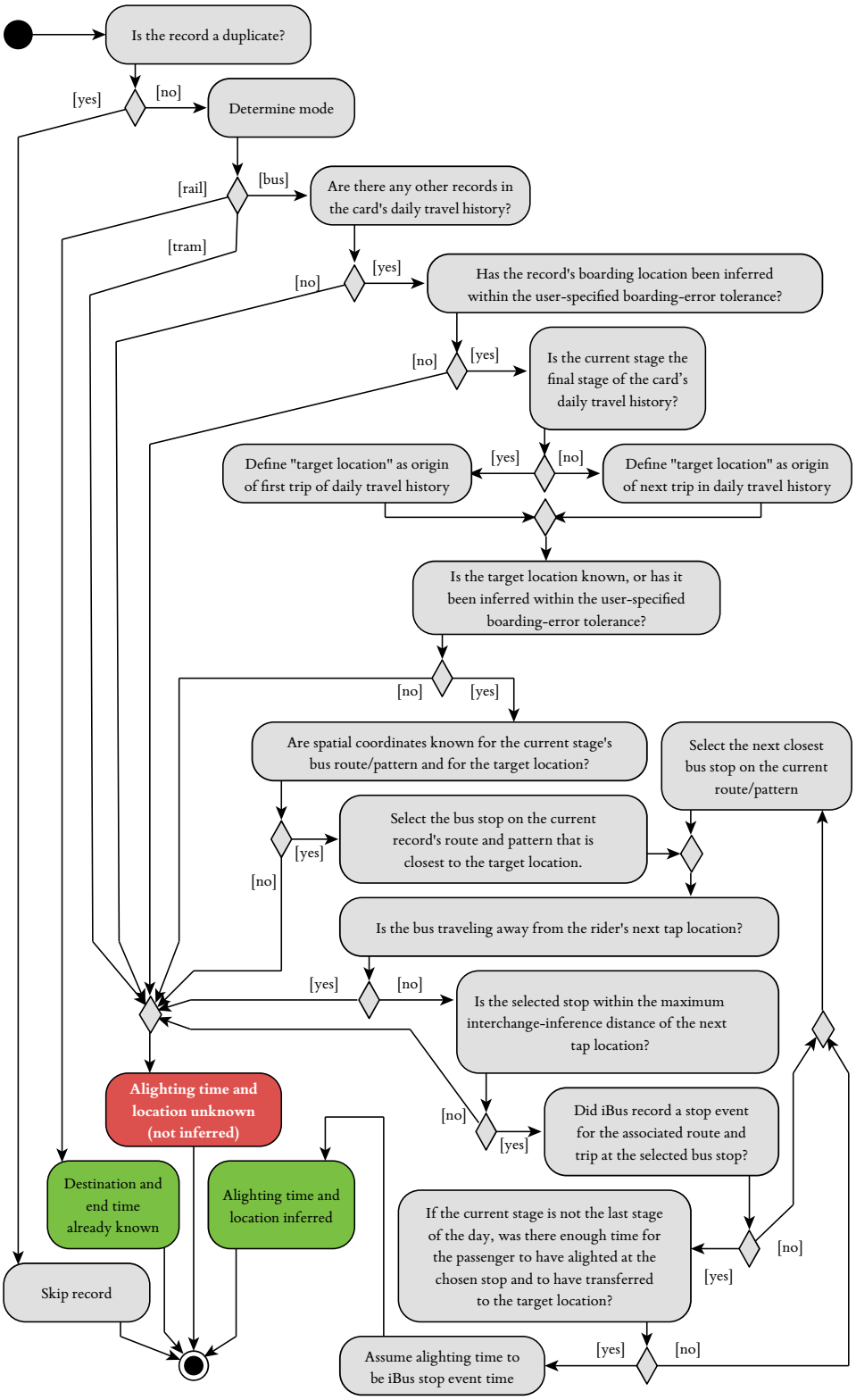


Figure 3-7. Activity diagram of the destination-inference process.

Next, the closest-stop and daily-symmetry rules are applied as illustrated in Figure 3-8. If a rider was inferred to have boarded a bus at stop 1, and if her next tap occurred at station Y, stop 5 would be selected as her tentative alighting location because it is the closest to Y. If her second tap occurred on another bus route rather than at a rail station, the same logic would apply: if her next journey stage was inferred to have originated at stop X, stop 4 would be selected as her tentative alighting location. If the current trip was the cardholder's last of the day, and if her target location (i.e., her first origin of the day) was stop 6, she would also be inferred to have alighted there for her final trip.

At this point, it is possible that the candidate alighting location is not a feasible destination for the cardholder. For example, if the cardholder was inferred to have started her current bus trip at stop 5, but if her subsequent transaction was a bus boarding at stop X, the bus was moving away from her subsequent boarding location: the closest stop (stop 4) was already served by the bus before she boarded. It is possible in this case that one or both of the origins were inferred erroneously, or that both origins are correct but that the rider did not alight at the closest stop to her next Oyster transaction, making the closest-stop rule inapplicable. To handle

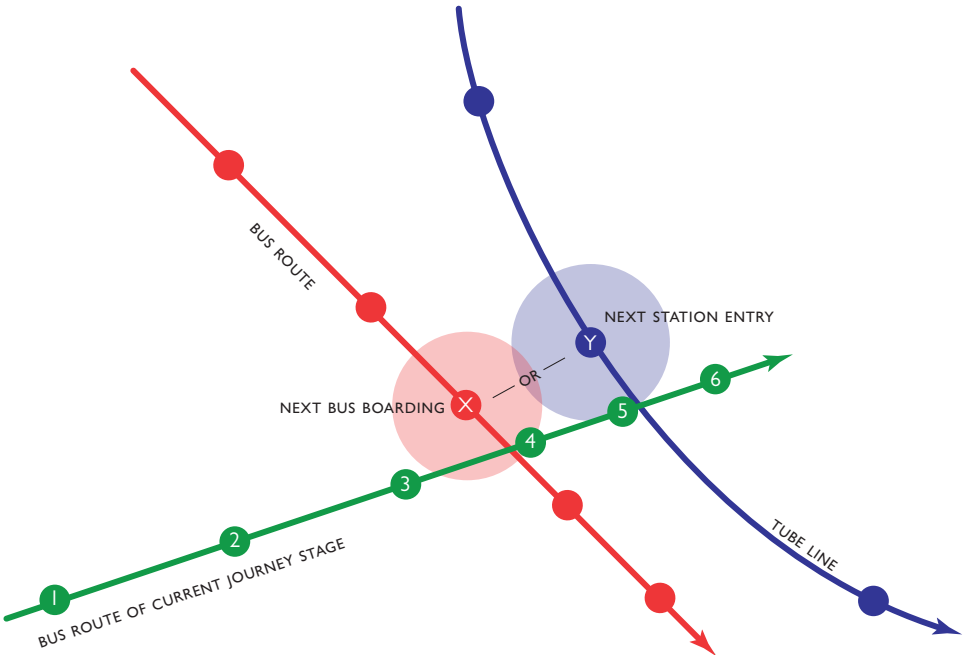


Figure 3-8. Example of destination inference for a passenger trip on a bus route (green) followed by another bus trip (red) or by a rail trip (blue).

this scenario, destinations are not inferred if the candidate alighting location is the same as its journey stage's boarding location, or if the candidate alighting location was served by the bus prior to its serving of the rider's boarding location.

The closest-stop rule is further checked against the distance between the candidate alighting node and the target node. If the closest stop on the current route is far from the cardholder's subsequent transaction location, it is assumed that proximity between the two stages was not a priority for the cardholder, and that the nearest-stop rule is therefore inapplicable. Thus, if the distance between the candidate alighting location and the target location is greater than the user-specified *maximum destination-inference distance*, the destination is not inferred.

If the candidate alighting node has passed all of the previous tests, an iBus record is sought for the bus route and trip (both of which are specified by the Oyster record) at the candidate alighting location. If found, the iBus record's arrival time is used as the candidate alighting time. If an iBus record cannot be found, the next-closest bus stop to the target node is chosen as the candidate alighting node, and testing is repeated starting with the determination of whether the bus was traveling away from the subsequent transaction.

Once a candidate alighting node passes all of the aforementioned tests, a final test is applied which checks whether the rider could have walked from the candidate alighting location to the target location in the available time. For example, if stop 4 in Figure 3-8 was chosen as an alighting location because it was the closest to stop X, but if the bus arrived at stop 4 several minutes after the rider boarded at stop X, it is likely that the rider alighted at an earlier stop. Stop 3 is then tested as the possible alighting location, because it was the next closest and next earliest, and therefore might have provided enough time for the rider to subsequently board at stop X. This test is not applicable for the rider's final trip of the day, since the duration of her egress is unknown. For any other transaction, however, the assumed minimum time required to move between the two locations, $t^*$, is calculated as:

$$t^{\star} = \frac{d}{\mathrm{R}} - \varepsilon$$

where

- $d$ is the Euclidean distance between the candidate alighting location and target location,
- R is the user-specified *maximum interchange speed*,
- $\varepsilon$ is the maximum boarding-error tolerance.

The maximum interchange speed is the greatest speed at which a person is assumed to be able to walk (or run) in the urban environment, taking into account that $d$ is the Euclidean rather than actual distance (i.e., the path taken is likely not straight), and that the pedestrian might be delayed by street crossings or other impediments. The maximum boarding error tolerance is then subtracted to account for the iBus arrival time not necessarily being the time at which the rider alighted. This parameter is used in this equation because it is assumed that the amount of error between a cardholder's alighting and its associated bus arrival event is generally similar to the amount of error between his boarding (and fare transaction) and its associated arrival event.

To test whether it was feasible for the rider to have alighted at the inferred location with enough time to walk to and tap at the subsequent location, her inferred alighting time, $t_1$, is compared to her subsequent transaction time, $t_2$, as follows:

$$t_1 + t^{\star} < t_2$$

If the expression is true, the passenger is inferred to have alighted at the specified location and time. But if the candidate time and location fail the test (i.e., if the expression is false), the next-closest stop becomes the candidate alighting location, and previous tests are repeated using the new candidate, starting with the determination of whether the bus was traveling away from the subsequent passenger origin.

### 3.3.3 *Destination-Inference Results and Sensitivity Analysis*

As discussed in section 3.2.3, the origin-, destination-, and interchange-inference processes were tested together using complete daily sets of Oys-

ter data for all of Greater London over ten consecutive weekdays.[14] Destination inference was completed within the first fifteen minutes of the program's execution (along with origin inference and part of the interchange-inference process). The number of destinations inferred, however, depends upon the setting of three user-defined parameters.

The first parameter, maximum walk speed, addresses to the speeds at which passengers travel when they are not in the public transport system (calculated as a function of time and the Euclidean distance traveled). Figure 3-9 illustrates the distribution of average speeds between bus or rail destinations and their subsequent rail origins.[15] The prevalence of average speeds below one kilometer per hour reflects passengers who performed activities between journey stages. Someone who was at his workplace for nine hours, for example, and who arrived and departed on the same route (having inbound and outbound stops on opposite sides of the street, 20 meters apart), would have an observed speed of 0.002 kilometers per hour.

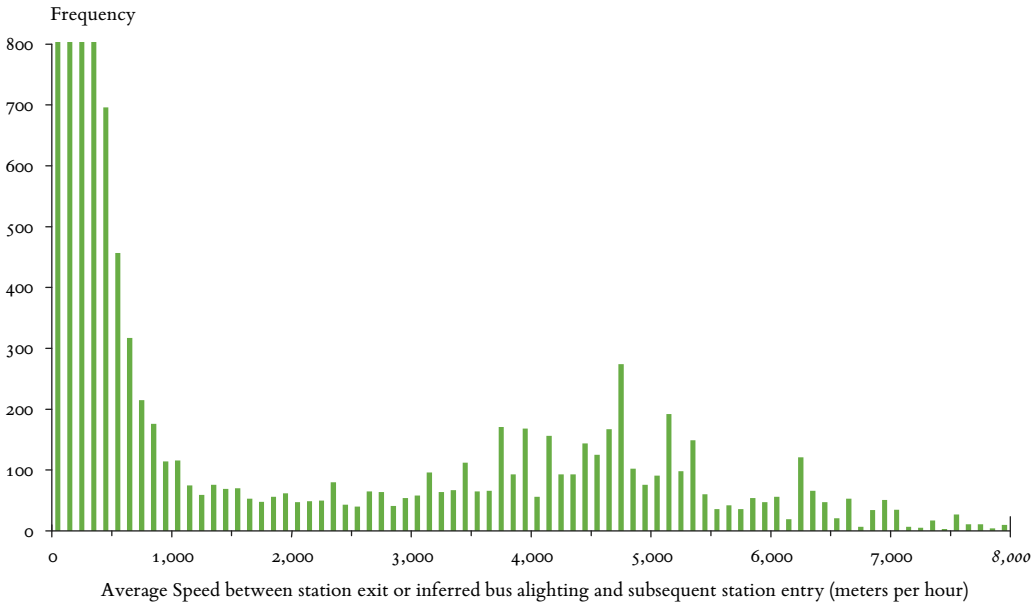


Figure 3-9. Distribution of average out-of-system speeds.

14 The data were obtained for the 6th through 10th and 13th through 17th of June 2011.

15 Speeds are calculated as Euclidean distances between station exits or inferred bus alightings and subsequent station entries. Subsequent bus boardings are excluded to eliminate wait time from the calculation.

The second most prevalent cluster of speeds is observed between 3 and 7 kilometers per hour, and likely reflects the speed at which passengers made interchanges.

Ninety percent of passengers exhibit out-of-system speeds of less than two kilometers per hour, which likely reflects activities such as work or school, or return trips that start where the previous trip ended. The 90th percentile speed is approximately 5 kilometers per hour, while 8 kilometers per hour reflects the 99.8th percentile. Since the maximum walk-speed parameter is used to test whether passengers had enough time to alight a bus before walking to their next transaction locations, the value of 8 kilometers per hours was chosen because it sets a conservatively high threshold for eliminating passengers from the destination-inference process. This speed is significantly higher than most of the speeds typically observed between Oyster trips, yet lower than any unreasonably high speeds that might indicate vehicle travel (perhaps by taxi, private automobile, or non-Oyster public transport).

The second parameter, boarding-error tolerance, is used by both the origin- and destination-inference processes. As described in the previous section, this amount of time is added to the maximum walk speed value to account for the errors typically observed between Oyster and iBus times. The sensitivity of this parameter and its setting of five minutes were described in section 3.2.3.

The final parameter used to estimate alighting locations is the maximum destination-inference distance. When the distance between a bus alighting location and the rider's subsequent fare transaction is great enough, it is reasonable to expect that the closest-stop rule does not apply. A distance of several kilometers, for example, could indicate that the rider took a non-Oyster transport mode between the two journey stages, and therefore did not necessarily alight at the stop closest to the subsequent fare transaction location. A more moderate distance such as 1.5 kilometers might have been traversed by foot, but the difference between any two consecutive potential alighting stops would be relatively small in comparison to the total distance walked by the cardholder during that segment of her journey, making its proximity to the next tap less important.

A maximum destination-inference distance should therefore be chosen that excludes longer values which might be less relevant to the closest-stop rule while retaining the shorter inter-stage distances which constitute the majority of observed Oyster activity. The distribution of

out-of-system distances is illustrated in Figure 3-10,[16] and a list of their cumulative frequencies is shown in Table 3-3. A value of 750 meters was chosen as the maximum destination-inference distance, which enables over 94 percent of observed bus stages to potentially be included in the results while avoiding the small marginal gains that would be realized by using a greater distance.

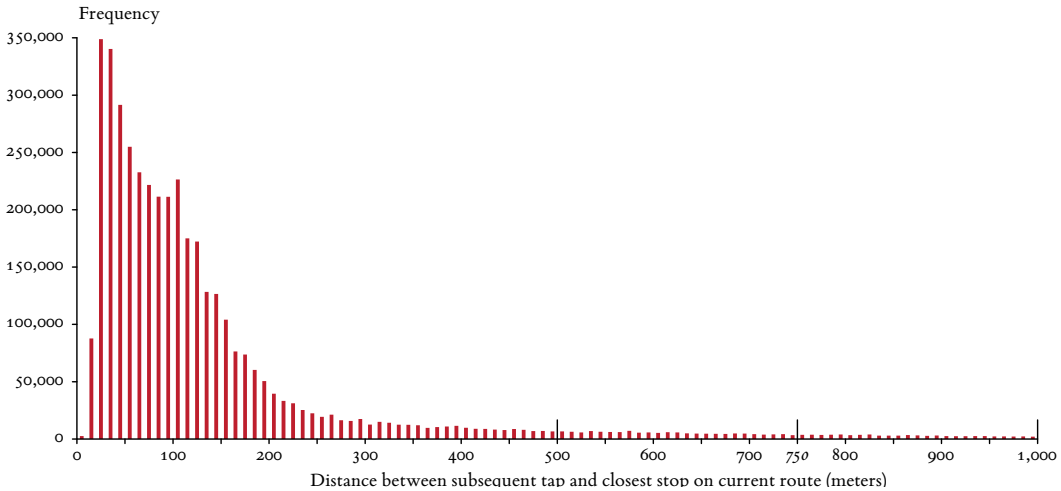


Figure 3-10. Distribution of Euclidean distances between candidate alighting stops and next fare transaction.

Table 3-3. Cumulative frequencies of observed inter-stage distances.

| Euclidean distance (meters) | Percentile | Marginal change |
|---|---|---|
| 500 | 90.5% | |
| *750* | *93.5%* | *3.5%* |
| 1,000 | 95.2% | 1.7% |
| 1,500 | 96.9% | 1.6% |
| 2,000 | 97.6% | 0.7% |
| 2,500 | 98.0% | 0.4% |

16 The candidate alighting stop is the closest stop on a cardholder's bus stage to his next Oyster transaction, regardless of whether that stop was eventually discarded by the destination-inference process in favor of another stop.

The rates at which bus alighting times and locations were inferred over the ten-day sample period are shown in Table 3-4.[17] The table also indicates the frequency with which passenger bus stages were excluded from the process by each of the algorithm's tests. Since failure of any one test can deem a record's destination "not inferable," tests are presented in the order in which they are applied: later tests can only exclude stages that have already satisfied the conditions of all earlier tests.

While most tests immediately eliminate transactions that do not satisfy their conditions, the two tests that are applied recursively are treated separately at the bottom of the table. If iBus data exist for the cardholder's route and vehicle but not for the candidate alighting node, or if there was not enough time for the rider to have alighted at the candidate location before her next transaction, the program resumes the destination-inference process using the next-closest stop. This process is repeated until a destination is successfully inferred, until one of the non-recursive tests is failed, or until all stops on the vehicle trip have been tested. The table indicates the frequencies at which the recursive tests were applied (counting each bus transaction only once, regardless of how many of its stops were tested). Most recursively tested bus stages were either successfully inferred or were eliminated by another test: only thirteen transactions per day typically exhaust all stops on the observed bus pattern during recursive testing.[18]

On average, destinations were inferred for 75.62 percent of Oyster bus stages, or roughly 4.7 million bus stages per day. Approximately 28 percent of the excluded bus stages (seven percent of total bus stages) could be retained by improving the origin-inference process or by widening its tolerance, while additional stages could be retained (at the expense of accuracy) by adjusting the maximum walk-speed and distance parameters.

---

17 The process was executed using a maximum boarding error of ±5 minutes, a maximum walk speed of 8 km/h, and a maximum destination-inference distance of 750 meters.

18 For example, if a rider boarded at the third to last stop, and if iBus events were not recorded for the last two stops, the transaction would exhaust all possible alighting locations and thereby fail all recursive tests.

Table 3-4. Destination-inference results:

| Result | Count | |
|---|---|---|
| DESTINATION NOT INFERRED BECAUSE: | | |
| Only one stage on card that day | 305,351 | 4.85% |
| Current stage's boarding location not matched to an iBus record | 80,707 | 1.28% |
| Current stage's boarding location inferred beyond allowable error (±300 sec.) | 156,580 | 2.49% |
| Next stage's boarding location not matched to an iBus record | 55,620 | 0.88% |
| Next stage's boarding location inferred beyond allowable error (±300 sec.) | 151,184 | 2.40% |
| Last stage of day; rider traveling away from first origin of day | 203,238 | 3.23% |
| Rider traveling away from next origin | 230,657 | 3.66% |
| Last stage of day; distance between candidate alighting location and first origin of day greater than 750 meters | 226,571 | 3.60% |
| Distance between candidate alighting location and next origin greater than 750 meters | 197,873 | 3.14% |
| All stops in pattern failed recursive tests (see below) | 12 | 0.00% |
| Subtotal, destination not inferred | 1,607,793 | 25.54% |
| Destination inferred | 4,687,842 | 74.46% |
| Total bus stages | 6,295,634 | |
| RECURSIVE TESTS: | | |
| iBus did not record an event for the nearest stop; tested next-closest stop | 54,593 | 0.87% |
| Not enough time for rider to have alighted at the nearest stop; tested next-closest stop | 20,424 | 0.32% |

## 3.4 SUMMARY

The methods presented in this chapter are shown to infer bus boarding locations in 96 percent of cases and of bus alighting times and locations in 76 percent. Most uninferred boardings are due to the Oyster and iBus data not matching within the five-minute threshold, although this parameter could be changed after additional validation. Further analysis should be conducted to test whether a more relaxed maximum boarding-error tolerance would be reasonable. The origin-inference tolerance doubly affects the destination-inference process, since origin locations are required both for the journey stage being processed and for the subsequent stage.

The software developed to execute this algorithm is discussed in detail in Chapter 6 and its outputs are used in the interchange-inference and full-journey matrix-expansion processes in Chapters 4 and 5. Applications of this algorithm, and of the algorithms that utilize its outputs, are presented in Chapter 7.

# Interchange Inference

4

The origin- and destination-inference processes discussed in the previous chapter enrich the Oyster data set by adding time and location information to most bus boardings and alightings. These enhanced bus records can then be analyzed along with Oyster rail transactions from the same card, enabling the observation of passengers' full journeys or daily travel histories.

This chapter presents a method for inferring whether each of an Oyster card's journey stages was linked to the next through an interchange (or transfer[1]). Doing so enables the analysis of full passenger journeys—including those that span multiple public transport modes—which are important descriptors of travel demand, and are critical for transport planning at the network level.

In addition to its value for providing full-journey information, this work is also useful for studying interchanges themselves. By yielding quantitative information about the distances and durations of interchanges—two key determinants of interchange disutility (Hine et al 2003)—this work can complement qualitative studies of interchange facilities to help planners improve the transfer experience, thereby reducing barriers to cross-modal travel and enabling cities and their residents to enjoy the social and economic benefits of increased mobility and accessibility (Guo and Wilson 2007, TfL 2001, 2002, 2009).

1 The terms *transfer* and *interchange* will be used interchangeably.

Before discussing the interchange inference process, it is necessary to define a few terms and concepts used in this research. First, since this chapter addresses the problem of inferring interchanges between AFC journey stages, it is necessary to clarify the concept of a journey stage. At many London Underground stations, for example, an Oyster cardholder can alight a train, walk to another platform (serving a different Underground line), and board another train without tapping out or leaving the station. By convention, this is informally referred to as an interchange, but the change is not recorded by the Oyster system because both of the customer's train rides occur between his entry and exit taps, thereby associating both trips with a single Oyster record.[2] In the context of this research, an Oyster journey stage (or simply *journey stage*) is therefore defined as any portion of a rider's travel activity that is represented by a single Oyster record. Each rail journey stage will therefore include one or more passenger trips, while the requirement that Oyster bus fares or passes be paid or validated onboard guarantees that each bus journey stage relates to a single passenger trip.

Second, it is necessary to define the concept of an interchange as it relates to this work. It should be generally agreed that the passenger in the previous example, or another passenger who alights a bus and immediately walks to, waits for, and then boards another bus (serving a route perpendicular to the first for example) both performed interchanges. Alternatively, it should be agreed that if the Tube passenger left the station and picked up his child from day care before returning to the station and boarding the second train, or that if the bus passenger collected a package at the post office before boarding her second bus, neither case would be considered an interchange. The distinguishing difference in both cases is the purpose for the transitions between the riders' trips. In the latter case, the passengers traveled to the locations mentioned in order to perform activities from which they derived utility (picking up a child and collect-

2 The Oyster data set actually contains two records for each rail transaction: one for the entry tap and another for the exit. These transactions are combined by the software prior to the interchange-inference process (as described in Chapter 6) and are thus considered a single record.

ing a parcel). In the interchange case, their purpose was simply to board another transit vehicle in order to derive utility from activities performed at subsequent destinations.

There are many cases, however, in which the discerning of interchanges from activities may be more subjective. If, for example, a passenger purchases a cup of coffee while walking from her alighting bus stop to her subsequent bus stop, she is deriving utility from her purchase but is unlikely to have traveled to that location solely for the coffee—she presumably could have made such a purchase at another interchange location or at her final destination. While this example might be considered both an activity and an interchange, the important distinction from a network-planning perspective is whether the customer chose to travel to that location to make her purchase, or made her purchase at that location because she happened to be transferring there. If the latter case were true it would mean that the customer's demand for travel lies with her ultimate destination rather than the bus stop near the coffee house.

For this research, an interchange is therefore defined as a transition between two consecutive journey stages that does not contain a trip-generating activity. The passenger may have derived utility from an activity performed during the transition, but the activity was not reason enough to make the trip: the primary purpose of the transition was to connect a previous stage's origin to a subsequent stage's destination.

Lastly, the inference of interchanges allows the linking of journey stages into full journeys. Taking into account the concept of journey stages and interchanges, it follows that a full passenger journey can be defined as a sequence of journey stages connected exclusively through interchanges. Passengers may still transfer between rail lines using a single Oyster transaction, but these "behind-the-gate" interchanges are included within rail journey stages and are therefore not considered in this algorithm.[3]

3 Behind-the-gate interchanges could be inferred by integrating a path choice model, such as those developed specifically for the TfL network (Guo 2008, Paul 2010) or elsewhere (Raveau et al 2010).

### 4.2.1 *Discerning Activities Using AFC Data*

This research aims to infer interchanges by distinguishing them from trip-generating activities, but activity information is not recorded by AFC systems. By noting the spatial and temporal characteristics of certain activity types, however, these patterns can be detected in the AFC data.

Holton (1958) distinguishes "convenience goods," for which proximity to the customer is a key concern, from "shopping" and "specialty" goods, for which variations in price, quality, or selection are great enough to warrant dedicated trips. Holton also notes that convenience purchases (generalized here to convenience activities) are typically shorter in duration than other types. The implications for this research are that convenience activities—which by definition are not trip-generating activities and which can therefore take place during interchanges—can be partially distinguished from trip-generating activities by their duration.

Kuhnimhof and Wassmuth (2002) mine a database of travel survey information to detect correlations between activity types and their spatial and temporal patterns such as duration, frequency, and time of day in order to calibrate the activity-generating component of a travel-demand forecasting model. Chu (2010) searches for similar spatial and temporal patterns in a set of AFC data from a bus network (for which he previously inferred origins, interchanges, and destinations) to infer general activity types such as home, work or school, and recreation.

### 4.2.2 *Interchange Inference Using AFC Data*

The tendency of trip-generating activities to be generally longer in duration than convenience activities has led several researchers to infer bus-to-bus interchanges according to the amount of time elapsed between consecutive boarding transactions or journey stages.[4] Bagchi and White (2004, 2005) assume that two bus stages are linked if they occur on different routes and if both AFC boarding transactions occur within 30 minutes of one another. Barry et al (2009) and the San Francisco Bay Area's re-

4 Thill and Thomas (1987) provide a literature review of early attempts at trip chaining.

gional planning organization (Metropolitan Transportation Commission 2003) use the same assumption, while Okamura et al (2004) apply a threshold of 60 minutes, McCaig and Yip (2010) allow 70 minutes, and Hoffman and O'Mahoney (2005) set an upper limit of 90 minutes.

Seaborn et al (2009) use Oyster boarding transactions to infer interchanges between London buses, and use Oyster station entry and exit data to infer transfers between bus and Underground. After exploring the distributions of time for these various journey types, they recommend thresholds of 20 minutes for tube-to-bus interchanges, 35 minutes for bus-to-tube, and 45 minutes for bus-to-bus. Seaborn et al then calculate the distributions of cardholder journeys per day and stages per journey, and compare both to the London Travel Demand Survey (LTDS).

Chu and Chapleau (2007, 2008) infer alighting times and locations using the GPS location-stamped boarding transactions of other cardholders, while Wang et al (2011) and Munizaga et al (2011) infer alighting times and locations using AVL data. All three studies thereby eliminate in-vehicle travel time from their observations, providing a more accurate estimate of actual interchange times. Munizaga et al set a 30-minute interchange threshold between bus stages, while Chu and Chapleau set time thresholds dynamically. By assuming a maximum walking speed of 4,320 meters per hour, they allocate an interchange time threshold as a function of the distance between the alighting and subsequent boarding locations, and add a five-minute buffer to prevent unreasonably short time thresholds when the two stops are extremely close together:

$$\textit{Interchange time threshold} = \frac{\textit{interchange distance}}{4{,}320} + 5$$

In addition to using time thresholds to test for interchanges, McCaig and Yip (2010) also apply a spatial condition. After classifying London's bus routes into four general travel directions (northeast, southeast, southwest, and northwest), they calculate the elapsed times between pairs of Oyster cardholders' bus boardings and group the results into the categories shown in Table 4-1.[5] Under the assumption that an interchange cannot take place if both transactions occurred on the same route, and testing the assumption that interchanges are not likely to include travel in the opposite direction, they show that the time distributions are consistent with these assertions (building upon the additional assumption that interchanges are likely to be shorter in duration than trip-generating activities). They then add a spatial condition to their proposed interchange-inference process by imposing the rule that an interchange cannot consist of two transactions in opposite directions.

Table 4-1. Median elapsed times between Oyster cardholders' consecutive bus boarding

| | Same Direction | Opposite Direction |
|---|---|---|
| Same Route | 85 min. | > 120 min. |
| Different Route | 30 min. | 55 min. |

## 4.3 METHODOLOGY

The interchange-inference algorithm developed in this thesis builds upon the research described in the previous section by applying a combination of spatial, temporal, and binary criteria to a complete set of AFC data collected from a multimodal transport network. Since these tests use tem-

5 Since McCaig and Yip measured the time between consecutive bus boarding transactions, the durations shown in the table include both the in-vehicle travel time of the earlier stage and the true inter-stage time (which includes any interchange, activity, or waiting time between the two stages).

poral and spatial data as proxies for passenger activity information, it is important to note that there are still certain trip-generating activities that could be mistaken for interchanges.

Returning to the example in section 4.1, if a rider alighted a bus, quickly picked up a package at the post office, boarded another bus serving a route perpendicular to the first, and finally alighted at her workplace, all of the methods discussed in section 4.2.2 would consider her to have made an interchange. Since the package was at that particular post office, there was a reason for the rider to visit that bus stop. Even if there were a direct bus from her origin (her home, for example) to her workplace, she would still have taken these two journey stages because she intended to visit that specific post office. In other words, the post office visit was a trip-generating activity and should not be considered an interchange.[6]

For this reason, each of the tests applied in this algorithm are used to determine whether a transition between two journey stages was *not* an interchange. Failing any one test will label the former transaction as not linked to the next, while any transactions that pass all tests can be considered *likely* interchanges. Since the previous example illustrates the possibility of producing false positives, the algorithm errs on the side of labeling transactions as not linked when interchange indicators are contradictory or ambiguous.

The algorithm requires two inputs: a set of origin- and destination-inferred Oyster data (the output of the OD-inference process), and a matching set of iBus data. The interchange status of each Oyster record is then inferred by applying the following binary, temporal, and spatial tests, as listed in Table 4-2 and illustrated in Figure 4-1. If any one test fails, the transaction is considered "not linked" to the following transaction: all further tests are skipped for that record and testing resumes with the next transaction. If a test fails because of missing data, the interchange status is considered to be "non-inferable." The record is considered unlinked

6 From an activity-based demand-modeling perspective, these two bus trips were part of the same *tour,* since they were not separated by a primary origin (e.g., home) or a primary destination (e.g., work), but were nonetheless two distinct single-stage journeys which included the post office visit (a trip-generating activity) in the tour. While the journeys inferred in this work could be grouped into tours, tours are extraneous to the interchange-inference process itself and are not considered further in this chapter.

rather than linked, but the algorithm notes its status as non-inferable for reporting purposes.

Table 4-2. Tests applied during the interchange-inference process

*Binary Conditions*
- Subsequent transaction
- Successful bus destination inference
- Complete rail exit information
- Repeat transport service
- Subsequent origin inference

*Temporal Conditions*
- Interchange time
- Maximum bus wait time
- Observed bus headway

*Spatial Conditions*
- Maximum interchange distance
- Circuity
- Full-journey length

### 4.3.1 *Binary Conditions*

*Subsequent Transaction*. The first condition tested is whether the transaction was the cardholder's final stage of the day. If so, it is considered unlinked.

*Successful Bus Destination Inference*. If the transaction represents a bus stage, its alighting time and location are required in order to apply the spatial and temporal tests. If a destination was not inferred for the transaction, its interchange status cannot be inferred.

*Complete Rail Exit Information*. Similarly, if the transaction represents a rail stage, its station exit time and location must be recorded. If not, the transaction cannot be inferred.

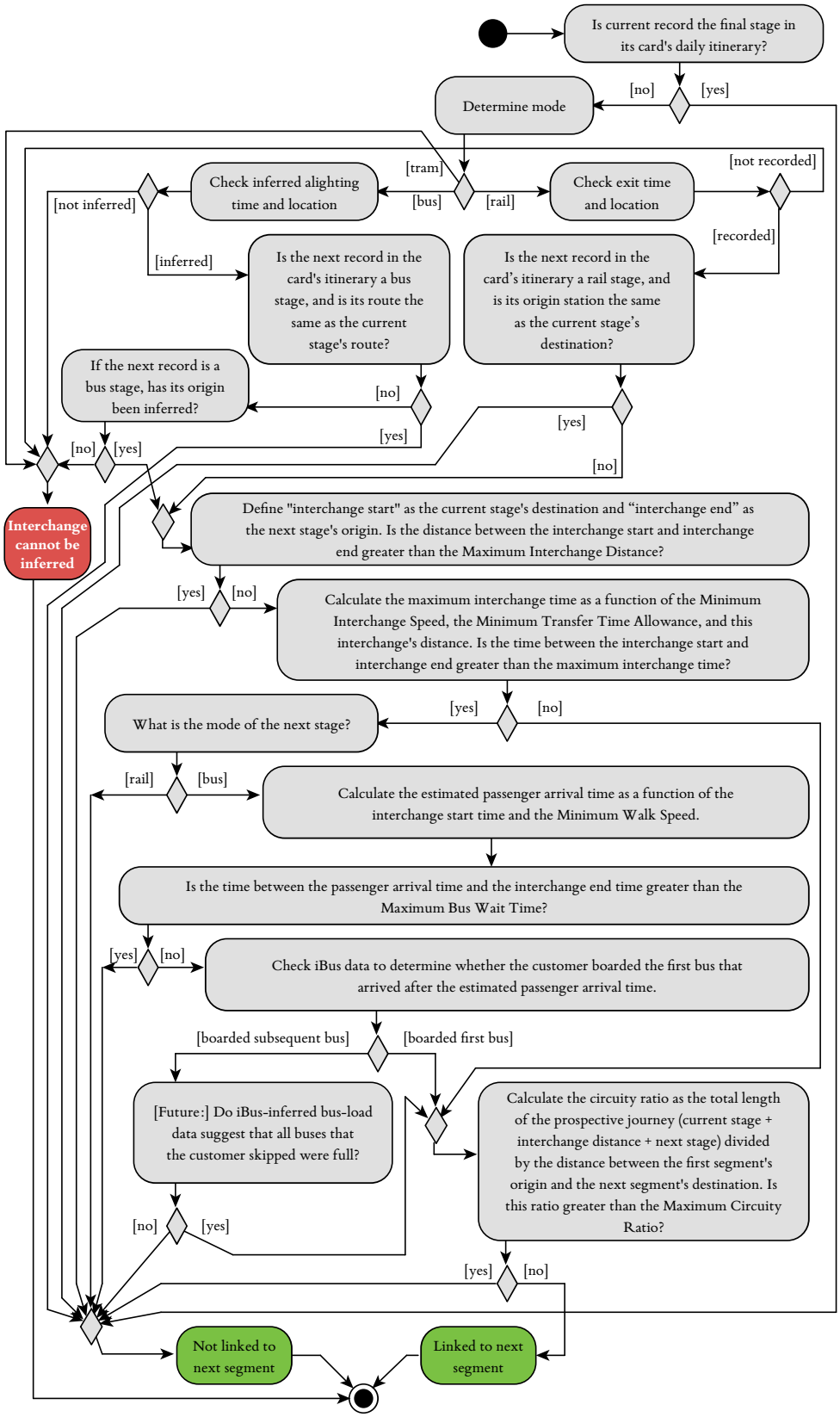


Figure 4-1. Activity diagram of the interchange-inference process

*Repeat Transport Service*. If two successive stages of an Oyster card used the same bus route or rail station, the first is not linked to the next, regardless of the direction of travel. Successive trips on the same service in opposite directions indicate a return trip, while travel in the same direction (on the same service) still indicates that a trip-generating activity was performed (otherwise the passenger would not have alighted the vehicle).

Since Oyster bus records denote the route, this test is applied by simply comparing the route numbers of the current and subsequent transactions. For rail transactions, the station IDs of the Oyster records are compared. If they match, the current record is considered unlinked.

*Subsequent origin inference*. If the record following the current transaction represents a bus stage without an inferred boarding location, it cannot be compared to the current record during the temporal and spatial tests. If a boarding location was not inferred for the following bus stage, the current stage's interchange status cannot be inferred.

### 4.3.2 *Temporal Conditions*

*Interchange Time*. Since it is assumed that trip-generating activities are likely to have longer durations than interchanges, a maximum interchange time is imposed. As with Chu and Chapleau (2008), this limit is calculated as a function of distance, but rather than adding a five minute buffer this algorithm sets a lower limit to the maximum interchange time, as follows:

$$t_{\text{int}} = \max \left[ t_{\text{min}}, \frac{d}{\text{R}_{\text{min}}} \right] \tag{4.1}$$

where

- $t_{\text{int}}$ is the maximum interchange time,
- $t_{\text{min}}$ is the minimum interchange-time allowance,
- $d$ is the interchange distance: the Euclidean distance between the current stage's alighting or exit location and the following stage's boarding or entry location,
- $\text{R}_{\text{min}}$ is the user-specified minimum walk speed,

The minimum interchange-time allowance is a user-defined parameter that provides a small buffer when short interchange distances would oth-

erwise result in unreasonably short interchange-time thresholds. For example, if using a minimum walk speed of 3,000 meters per hour, two intersecting bus routes with a pair of stops 20 meters apart would otherwise allow only 24 seconds in which to transfer. The minimum interchange-time allowance parameter allows for errors between Oyster and iBus timestamps, while providing a small buffer for non-trip-generating activities such as reloading an Oyster card.

If the next record represents a rail stage, and if the time between the current stage's exit or alighting time ($t_1$) and the following stage's station-entry time ($t_2$) is greater than the maximum interchange time ($t_{int}$), it is assumed that the cardholder was engaged in an activity and the current stage is considered not linked to the next. If the following record is a bus stage, however, this test is applied and its result temporarily recorded, but the current record's interchange status is not affected, since the time between the two stages might include some amount of bus wait time. In such cases, the result of this test will be taken into account during the two remaining temporal tests.

*Maximum Bus Wait Time*. This test is only applied if the following record represents a bus stage and if the time between stages was greater than the maximum interchange time ($t_{int}$) in the previous test. If both conditions are met, it is assumed that the rider either waited at the bus stop or was engaged in a trip-generating activity before boarding.[7] It is assumed that there is some upper limit to the amount of time that someone will spend waiting for a bus, regardless of the bus's headway. The estimated time between the passenger's arrival at the stop and her subsequent boarding is therefore calculated as $t_2 - (t_1 + t_{int})$, and if the result is greater than the user-specified maximum bus wait time parameter, the current stage is considered not linked to the next.[8]

---

7 It is also possible that the rider boarded the bus immediately upon arrival at the stop, but for consistency, riders are given the same time allowance to walk to a bus stop as they are to walk to a rail station.

8 $t_{int}$ is used in the maixmum-bus-wait-time test for the same reasons as in the interchange-time test: short interchange distances are guaranteed a minimum time allowance to account for the difference between Oyster and iBus times, or to allow brief, non-trip-generating activities.

*Observed Bus Headway*. This test is only applied if the current record passed the maximum interchange test. In other words, the following transaction represents a bus stage and the cardholder boarded the bus after the allotted maximum interchange time but before the maximum bus wait time. It might therefore be assumed that the customer waited at the stop before boarding the bus, but by searching the iBus data the program can determine whether another bus assigned to the same route served the stop while the customer was presumably waiting. If the rider did not board the first bus that served the stop during this period, it is assumed that he was engaged in an activity rather than waiting.[9] The iBus data are searched to determine whether another vehicle served the route specified by the rider's following trip during the interval $[t_1 + t_{\text{int}}, t_2)$, and if so, the current stage is considered not linked to the next.

### 4.3.3 *Spatial Conditions*

*Maximum Interchange Distance*. Since the primary purpose of an interchange is to transition from one vehicle to another (and to ultimately perform some activity at a subsequent destination), it follows that there should be some limit to the distance between the current stage's alighting or exit location and the following stage's boarding or entry location if the transition is to be considered an interchange. The Euclidean distance between the two points is compared to a user-specified maximum-interchange distance, and if the cardholder's interchange difference is greater, the current trip is considered not linked to the next.

*Circuity*. In addition to a maximum interchange distance, it is also assumed that a multi-stage journey will not entail an overly circuitous path. If the Euclidean distance traveled over two consecutive stages is sufficiently greater than the Euclidean distance between the current stage's origin and the next stage's destination, it is assumed that the disutility of the additional travel would outweigh the utility of the interchange.

---

9 The rider might have skipped the first bus because the vehicle was full or because the driver did not stop (for example, if the driver knew that another bus was close behind). These criteria are being considered for a future version of the algorithm but in the present version it is assumed that, over the course of the entire day and network, it is more likely that a skipped bus indicates an activity.

The algorithm tests for excessive circuity by first calculating the Euclidean distances traveled during the current stage ($d_{cur}$), the following stage ($d_{nxt}$), between the current stage's destination and the following stage's origin ($d_{int}$), and directly between the current stage's origin and the following stage's destination ($d_{dir}$). The sum of the first three distances is considered the distance traveled, while the third is considered the direct distance. The ratio of these distances, $(d_{cur} + d_{int} + d_{nxt}) / d_{dir}$, is compared to the user-specified *circuity ratio* parameter, and if the observed ratio is greater than the parameter, the current stage is considered not linked to the next.

The benefit of this metric rather than an angular cost is that, by taking distance into account rather than deviation, reverse travel is permitted but penalized. Figure 4-2 shows two potential interchange locations for the same origin and destination (assuming a zero interchange distance for simplicity). Both paths have the same cumulative Euclidean distance, but in the example to the right the destination is farther from the interchange location than it is from the origin: the rider spent the first stage traveling away from the second stage's destination.

The ratio of the distance traveled to the direct distance defines an ellipse with the first stage's origin and the second stage's destination as its foci. These elliptical bounds are more tolerant of lateral travel than of reverse travel, which is useful for defining interchanges. While a passenger might travel backwards for a short distance—for example, to take a slower but more frequent local service before transferring to a faster limited-stop service—it is assumed that riders are not likely to travel as far backward as they are laterally.

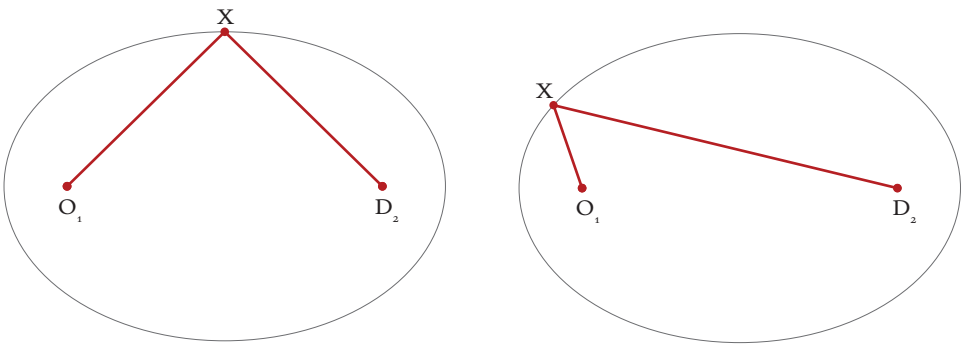


Figure 4-2. Two potential interchange locations yielding equal cumulative Euclidean distances

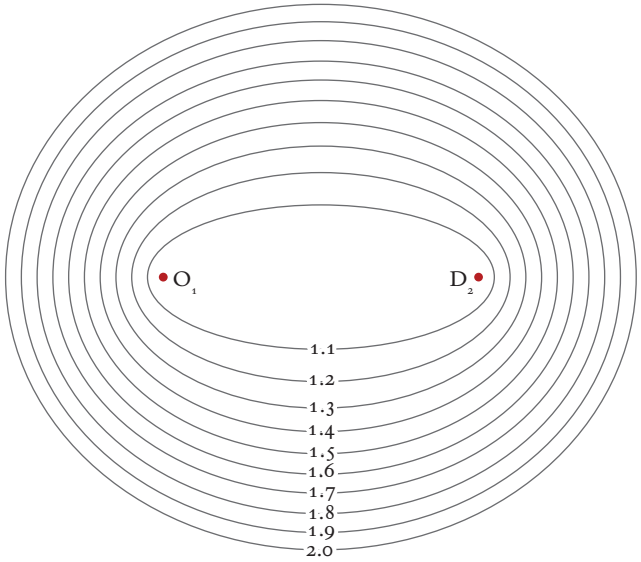


Figure 4-3. Interchange boundaries for various circuity ratios.

*Full-Journey Length*. While the circuity test helps to prevent round trips from being erroneously classified as single journeys, it only takes two successive journey stages into account. After all other tests have been applied, full journeys are tentatively defined by grouping stages according to their inferred link status. It is possible, however, that a set of journeys that includes a return journey (which by definition ends sufficiently close to the journey's starting point) passed the circuity test because no single transition was excessively circuitous, but the cumulative angular change over two or more transitions resulted in a return journey. The journey-length test therefore "unlinks" all stages in the journey (by reclassifying each stage as not linked to the next) if the journey's origin and its destination were closer than the user-defined *minimum journey length* parameter.[10]

10 Two-stage journeys can be unlinked in this way but, in the case of journeys with three or more stages (hence two or more interchanges), it is possible that only one of the tentative interchanges must be unlinked in order to satisfy the journey-length condition, but it is unclear which one. To infer interchanges as conservatively as possible, all of a journey's links are broken in such cases.

The interchange-inference algorithm was incorporated into the same Java application that executes the origin- and destination-inference processes. The software was tested on the same ten-weekday period used in Chapter 3,[11] and typically completed in less than 20 minutes for each day, which includes the time taken to execute the previous two processes. The application creates a copy of the Oyster data set enriched with the fields inferred by the three processes. This section discusses the sensitivity of the algorithm's parameters and then presents the results of the ten-day analysis.

### 4.4.1 *Sensitivity Analysis*

The results of the interchange-inference process are influenced by the values of the following six user-defined parameters, which allow the user to select a balance between the accuracy and completeness of the results:

- Minimum walk speed
- Minimum interchange time allowance
- Maximum interchange distance
- Maximum bus wait time
- Circuity factor
- Minimum linked-journey distance

This section explores the sensitivity of the results to these parameters by observing the distributions of the associated properties in the ten-day Oyster data set.

*Minimum Walk Speed*. Figure 3-9 (page 53) illustrates the distribution of out-of-system speeds recorded between consecutive journey stages, where the latter transaction represented a rail stage (in order to exclude bus wait time from the calculation). As discussed in section 3.3.3, the speed is calculated as a Euclidean distance and therefore does not take the cardholder's path into account. The primary peak near 0 kilometers per

11 The 6th through 10th and 13th through 17th of June 2011.

hour likely represents activities while the secondary peak just below 5 kmph presumably indicates interchanges. Since the minimum walk speed parameter sets a floor below which stages will be considered not linked, it should be set low enough to retain people who may move slowly yet high enough to exclude shorter trip-generating activities. The tests run in this analysis used a value of three kilometers per hour, which appears to provide a reasonable tradeoff between the two criteria and excludes 96.4 percent of the distribution, most of which represents obvious activities. Decreasing the parameter to 1.9 kilometers per hour reduces the exclusion rate by one percent (to 95.4 percent), while raising it to 3.7 increases the rate by one percent (to exclude 97.4 percent).

*Minimum Interchange Time Allowance*. When potential interchanges are tested for excessive interchange time, this parameter provides a floor to the criterion. It should be set in a way that enables the test to exclude transitions of excessive duration while preventing the imposition of any unreasonably short time limit.

Figure 4-4 shows the cumulative distributions of inter-stage time for four stage-transition categories (all four permutations of bus and rail stages, measured from the prior stage's exit or inferred alighting time to the latter stage's entry or boarding time). Since most rail-to-rail interchanges occur behind the gate (and therefore within an Oyster journey stage), the distribution of rail-to-rail times is likely to contain mostly ac-

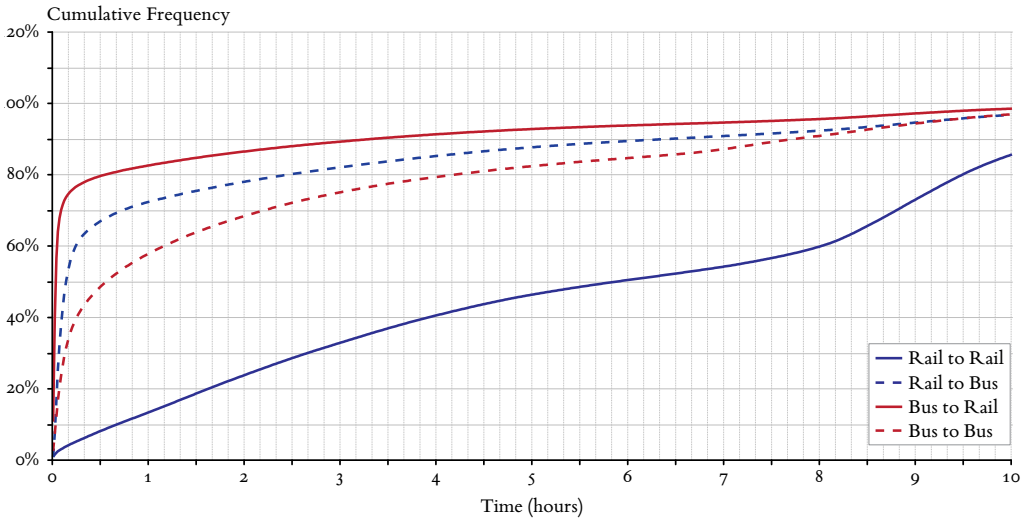


Figure 4-4. Time between cardholders' successive journey stages, ten-day average.

tivities and few interchanges. The notable rise between 8 and 9.5 hours likely represents workdays, which supports this assumption.

The three other distributions likely contain a mix of interchanges and activities. Since bus-to-bus transitions can include return stages on the same route, they are likely to contain a higher proportion of activities than the remaining two distributions (the bus-to-bus distribution also includes bus wait times). The rail-to-bus distribution has a higher proportion of shorter transitions than the bus-to-bus distribution, likely due to the absence of return trips on the same service.

The bus-to rail distribution, like rail-to-bus, is likely to contain a higher proportion of interchanges than the first two distributions. However, bus-to-rail transitions exclude wait time, making this distribution a more appropriate indicator of interchange times.

The 70 percent of bus-to-rail transitions that occur within five minutes likely indicate interchanges, as many bus routes in London stop very close to rail stations. The tapering at approximately 12 minutes presumably indicates longer interchange distances, yet these can be accommodated by the program because an interchange time threshold is applied as a function of distance. Setting a minimum allowance of five minutes only affects those cases where the interchange occurs over a very short distance, but ensures that the 70 percent of transitions that occur within that time limit (and which presumably are mostly interchanges), are protected against being erroneously considered not linked. Decreasing this parameter below five minutes can dramatically affect the inference rate of short-distance interchanges, while doubling the setting to ten minutes only changes the potential output from 70 to 75 percent.

*Maximum Interchange Distance*. Figure 3-10 (page 55) shows the distribution of distances between Oyster stages while Table 3-3 (page 55) reveals the sensitivity of the maximum destination-inference distance. The same distribution and sensitivity apply to the maximum interchange distance, and a value of 750 meters is therefore applied to this parameter as well.

*Maximum Bus Wait Time*. While Figure 4-4 shows the inter-stage times for all Oyster transactions, Figure 4-5 combines bus-to-bus and rail-to-bus times, after subtracting the estimated interchange time required for the rider to arrive at the stop (assuming tentatively that the transition might be an interchange). The dense but tapering cluster of times at the left of the distribution fits the expected pattern of passenger wait times resulting from the daily average of headways over the entire bus network, while the long tail at the right should represent activities.[12] While a threshold of 30 minutes might seem a reasonable cutoff for discerning waiting times from activities, the observed headway test provides a more robust measure of how long a customer might actually have waited. For this reason, a maximum bus wait time of 45 minutes was applied in this research, providing a backup in the presumably unlikely event of bus headways exceeding that duration (for example, during service disruptions). This conservative estimate excludes the 42 percent of the distribution that occurs to its right and which is very unlikely to contain a significant amount of interchange activity, while allowing the observed-headway test to act upon 58 percent of the distribution to its left. When set to 45 minutes, a change in the maximum bus wait-time parameter of plus or minus three minutes corresponds to a one percent change in the number of transactions eliminated by the maximum-bus-wait-time test.

12 The typical headways for London bus routes should result in a large number of customers waiting for only a few minutes, which is likely represented by the high frequency of short wait times in the histogram. Activities, by contrast, should be expected to exhibit a far greater range of durations—for example, an hour for a meal or more than eight hours for a work shift. Activities are therefore likely to be spread sparsely over the long tail of the distribution.

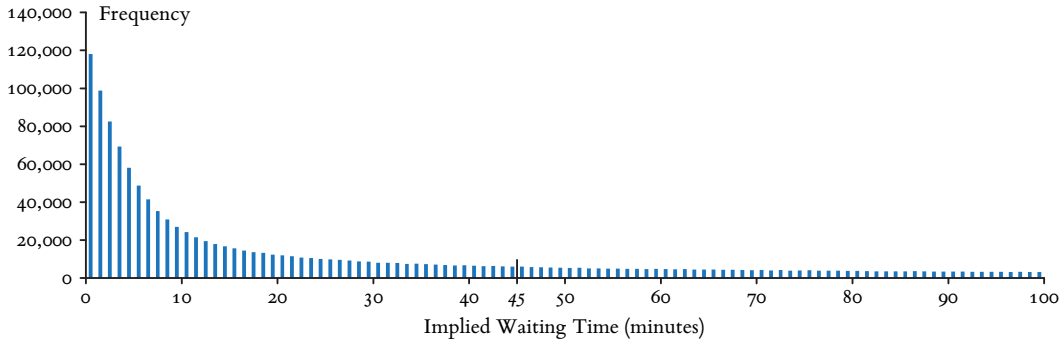


Figure 4-5. Histogram of elapsed times between candidate bus-stop arrival times and subsequent bus boarding times.

*Circuity Factor*. Figure 4-6 shows the distribution of circuity ratios for all potential interchanges that passed all previous tests (all binary and temporal tests, and the maximum interchange-distance test). More than half of the distribution lies below 1.07, and the 95th percentile has a factor of 1.8.

The concentration of low circuity ratios is likely due in part to the spatial arrangement of National Rail terminals in Central London. These stations encircle the urban core, leading many commuters from outside the city center to travel most of their journey by National Rail before transferring to a TfL mode for the relatively short final stage of their trip. For example, Figure 4-7 shows that a customer traveling by National Rail (green) from Honor Oak Park to London Bridge, then transferring to the Jubilee line (gray) and arriving at Bond Street, would travel within the bounds of the 1.1 circuity factor shown on the map. Many National Rail stations are much farther from the core than Honor Oak Park, leading to even lower ratios.

Despite the dense concentration of circuity ratios below 1.1, the algorithm was applied with a factor of 1.7. Figure 4-8 shows a shorter journey, from London Fields National Rail station to Angel Underground station. If this journey were made by traveling north to Hackney Downs and transferring to the North London Line (orange) at Hackney Central, it would require a circuity ratio of at least 1.5 (inner ellipse). If the rider instead traveled south and made the interchange at Liverpool Street station, a circuity ratio of 1.7 (outer ellipse) would be needed. Although few journeys have such high circuity factors, these journeys (and similarly circuitous bus journeys) are feasible commutes and should not be excluded.

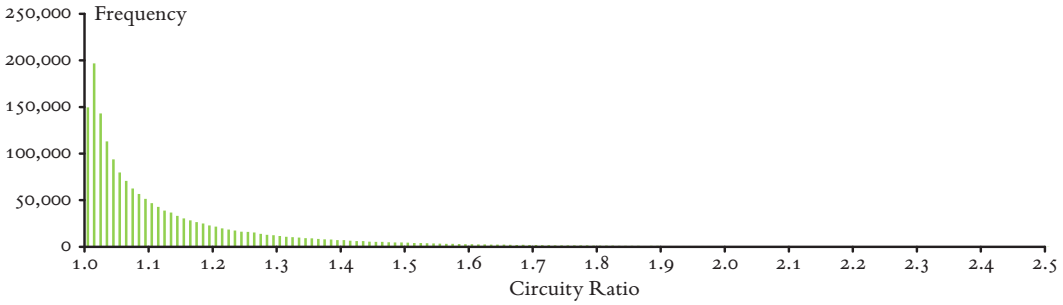


Figure 4-6. Histogram of circuity ratios for potential interchanges (inter-stage transitions that passed all previous tests).

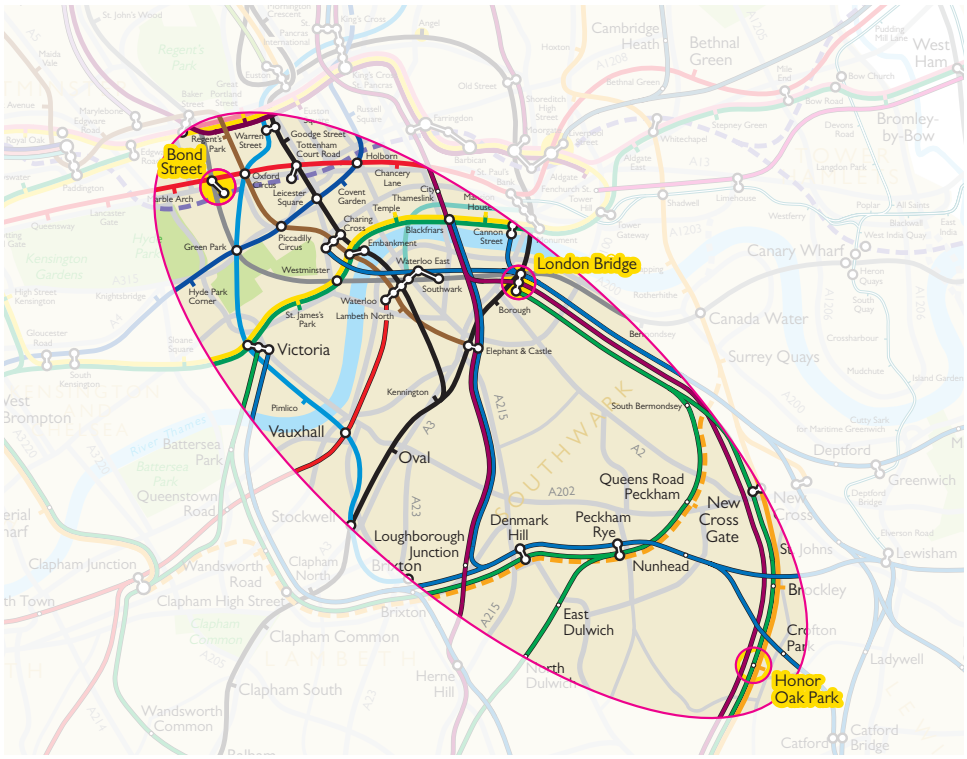


Figure 4-7. Honor Oak Park to Bond Street via London Bridge: Circuity factor of 1.1 (base map: TfL).

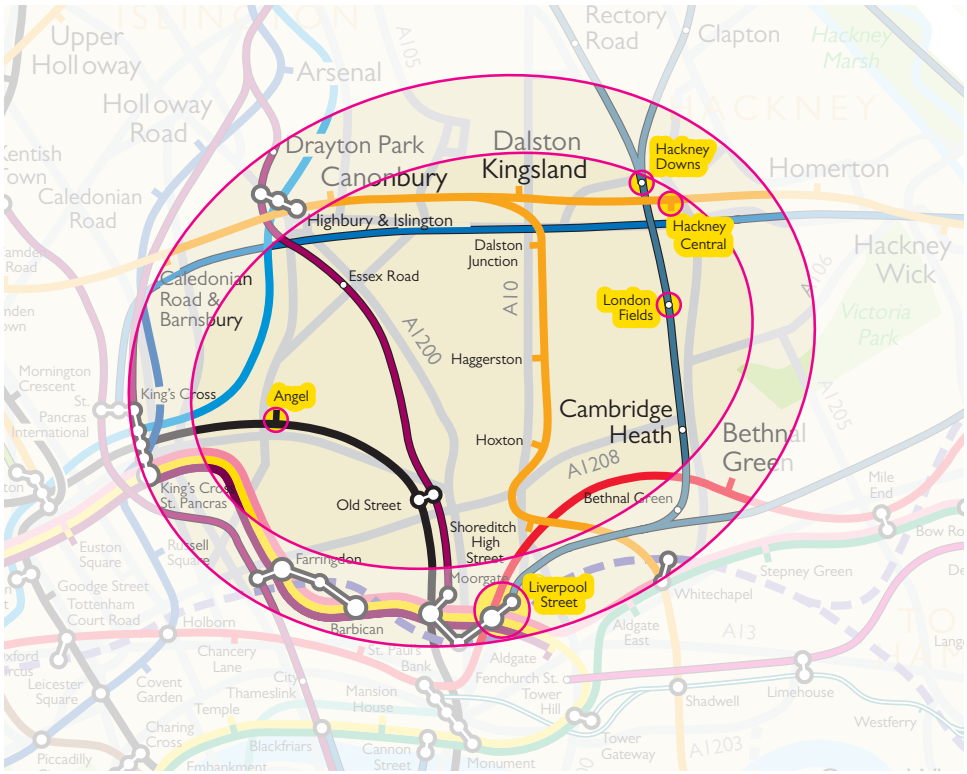


Figure 4-8. London Fields to Angel via Hackney or Liverpool Street: Circuity factors of 1.5 and 1.7 (base map: TfL).

*Minimum Linked-Journey Distance*. Figure 4-9 shows the distributions of Euclidean distances between the start and end points all potential full-journeys before applying the minimum journey-distance test. The small cluster at the left of the distribution presumably represents sets of two or more journeys that contain round trips, but which were not eliminated by the previous tests. The transition between the two clusters appears to occur at 275 meters, but a more conservative limit of 400 was applied, since common experience suggests that distances any shorter are unlikely to be traveled on linked journeys.

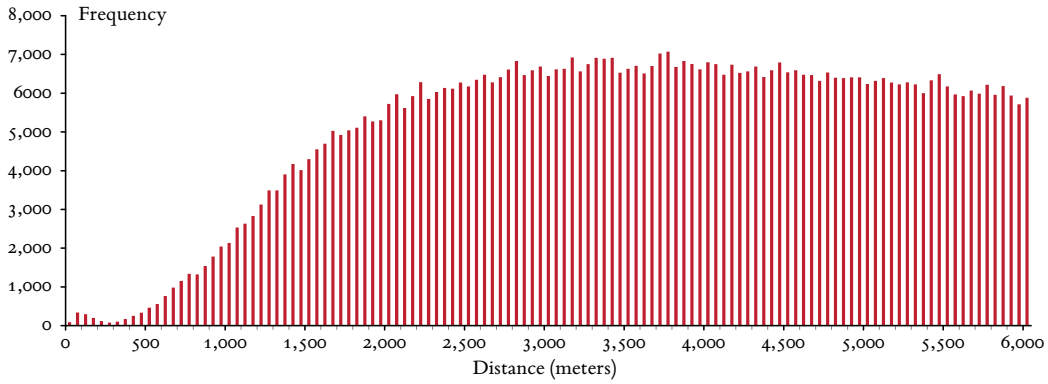


Figure 4-9. Histogram of potential full-journey distances.

### 4.4.2 *Results*

The interchange-inference process was performed on the ten-day Oyster sample using the parameters in the previous section, yielding the results shown in Table 4-3. Since bus origin and destination information is required when inferring interchanges to, from, or between bus stages, the inference rates of the origin- and destination-inference process affect the interchange-inference rates of bus-to-bus, bus-to-rail, and rail-to-bus transactions.

The number of Oyster cards recorded over the ten day period ranged from 3.0 to 3.1 million per day, containing 9.5 to 10.1 million daily journey stages linked into 6.6 to 7.1 million daily journeys. The median journey duration was 24.5 minutes, with the distribution shown in Figure 4-10.

Table 4-3. Interchange-inference results: ten-day average.

| Result | Count | Percentage |
|---|---|---|
| LINK STATUS CANNOT BE INFERRED BECAUSE: | | |
| Current stage is tram | 38,776 | 0.39% |
| Current boarding location not inferred | 80,707 | 0.82% |
| Current boarding location inferred beyond maximum error | 156,580 | 1.58% |
| Next stage's boarding location not inferred | 55,620 | 0.56% |
| Next stage's boarding location inferred beyond maximum error | 151,184 | 1.53% |
| Bus destination not inferred | 140,248 | 1.42% |
| Bus destination inferred beyond maximum error | 197,873 | 2.00% |
| Next's stage's bus destination not inferred | 90,409 | 0.91% |
| NOT LINKED TO NEXT BECAUSE: | | |
| Final stage of day | 2,955,332 | 29.89% |
| Same bus route | 886,344 | 8.96% |
| Same rail station | 1,034,640 | 10.46% |
| Beyond maximum interchange distance | 276,022 | 2.79% |
| Walk to rail stage slower than minimum walk speed | 334,111 | 3.38% |
| Maximum bus wait time exceeded | 835,178 | 8.45% |
| Did not board first observed bus | 363,741 | 3.68% |
| Too circuitous | 109,395 | 1.11% |
| Less than minimum journey length | 2,481 | 0.03% |
| Subtotal, interchange not inferred | 911,397 | 9.22% |
| Subtotal, not linked to next | 6,797,244 | 68.75% |
| Subtotal, assumed linked to next | 2,139,578 | 21.73% |
| Total stages | 9,848,219 | |

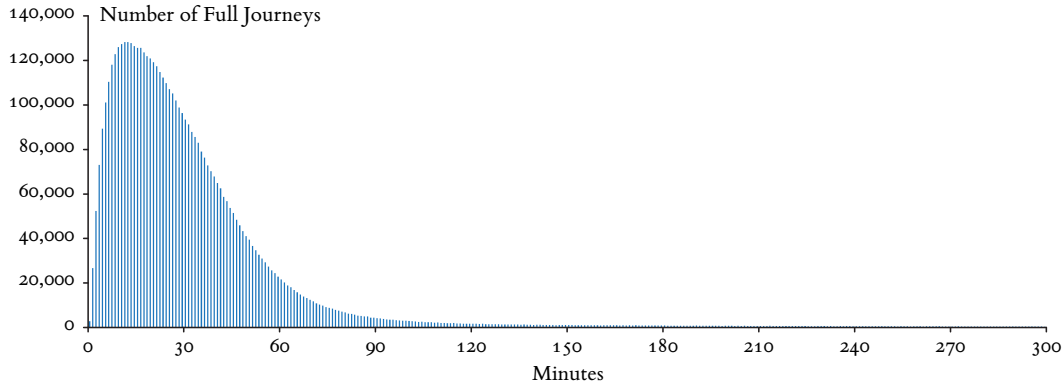


Figure 4-10. Histogram of inferred journey durations.

### 4.4.3 *Comparison with the London Travel Demand Survey*

Following Seaborn et al (2009), the results of the interchange-inference process were compared to the London Travel Demand Survey (LTDS) of the three-month period containing the ten-day Oyster study period.

A comparison of the number of daily journeys per passenger is shown in Figure 4-11. Consistent with the findings by Seaborn et al, the survey indicates a significantly higher proportion of passengers with two journeys per day than the Oyster data. This might be due to sampling error (the LTDS sample is much smaller than the Oyster data set) or bias—for example, respondents might be more likely to report their journeys to and from work or school while underreporting recreational, unplanned, or infrequent travel.

The difference between the two distributions might also indicate an underestimation of interchanges by the algorithm. Table 4-3 shows that 9.2 percent of journey stages could not be inferred, and these stages are treated as single-stage journeys (rather than discarding them and leaving gaps in cardholders' daily travel histories). Furthermore, the decision to apply the interchange-inference criteria conservatively may have caused some number of multi-stage journeys to be erroneously classified as multiple single-stage journeys, which might explain the larger number of journeys per day in Figure 4-11, and, conversely, the smaller number of stages per journey in Figure 4-12.

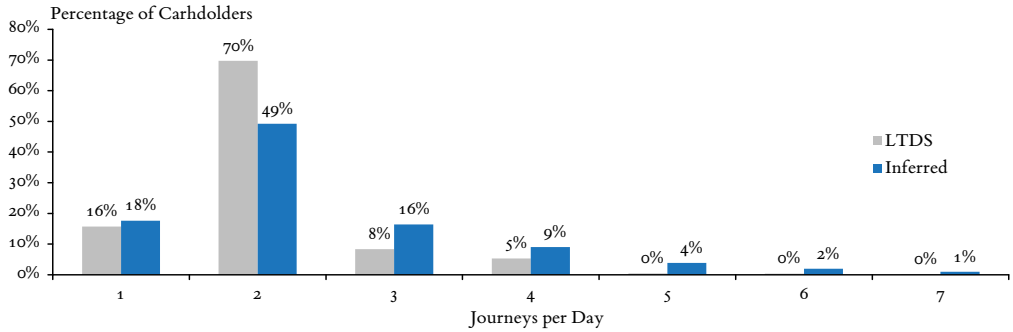


Figure 4-11. Full journeys per passenger per day.

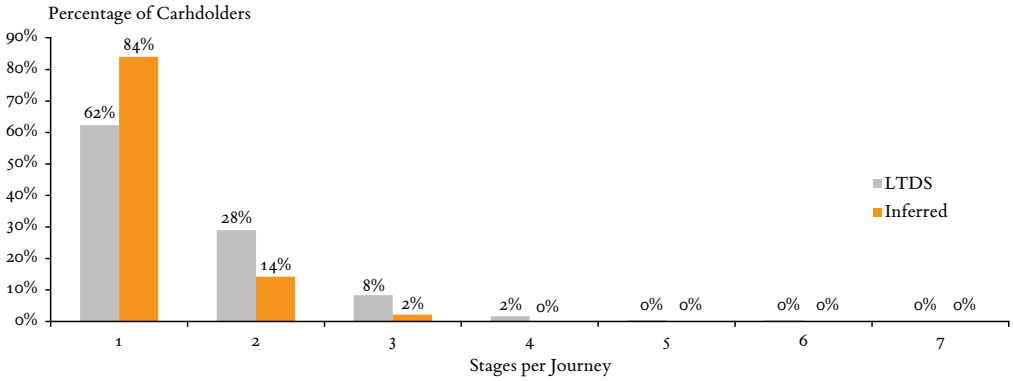


Figure 4-12. Stages per journey.

## 4.5 SUMMARY

The algorithm presented in this chapter employs a range of temporal, spatial, and logical assumptions about public transport interchanges and applies a series of tests to infer whether each journey stage in a set of roughly 9.8 million is linked to its cardholder's following stage. Stages of various public transport modes can then be joined to reconstruct the full journeys and daily travel histories of roughly 3 million daily customers.

The tests for inferring interchanges were applied conservatively, and a comparison to the results of London's principle travel survey suggests that the algorithm, when applied using the parameters described in this chapter, might be underreporting transfer activity. The interchange-inference

rate could also be increased by improving the inference rates of the origin- and destination-inference processes, which provide its inputs.

Further studies of the algorithm's outputs could aid in the tuning of its parameters. By mining multiple days worth of data, for example, activity types could be inferred, potentially revealing correlations between a trip's purpose and its spatial and temporal attributes. Different values for the six interchange-inference parameters could also be applied to different zones of the city, different service types, or during different times of day.

The implementation of the interchange-inference algorithm is discussed in Chapter 6, and multiple examples of its application are presented in Chapter 7.

# Full-Journey Matrix Expansion

5

The methods described in previous chapters infer the details of passengers' boarding, alighting, and interchange activity, which in turn yield information about passengers' full journeys. Although these methods utilize complete populations of AFC data, some AFC journeys are not properly recorded or cannot be inferred, while some riders pay their fares or provide proof of payment using other media. The large market share of London's Oyster card and the high inference rates of the aforementioned methods result in a very large sample of full passenger journeys (approximately 70 percent of all TfL passenger activity is inferred), but analyses of aggregate travel behavior require that the sample values be scaled to estimate the complete population of passenger activity on the TfL network.

This chapter describes a method for estimating expansion factors for full passenger journeys during one (or more) user-defined time periods, enabling the construction of a passenger flow matrix of travel activity across multiple public transport modes. The method is then validated by comparing the journeys' constituent rail-stage flows to those estimated using a traditional single-mode OD matrix of the Oyster rail network.

## 5.1 PROBLEM DEFINITION

Flows of passengers, vehicles, or goods are often described using origin–destination (OD) matrices. By constructing a table in which each row representing an origin and each column a destination, the value in each cell can be used to indicate the flow between a distinct OD pair. This is useful

for describing passenger flows between any two points during a given time period, where the origins and destinations could be the entry and exit stations on a closed rail network,[1] the boarding and alighting stops on a single bus route, or the geographic zones connected by a highway network. Full passenger journeys could be studied in a similar fashion—for example, by counting the number of passengers who start at a given bus stop and end at a given rail station, regardless of their intermediate transfer locations—but the inclusion of interchange locations in this work necessitates a different analytic framework.

Rather than estimating a scaling factor for each OD pair, this work seeks to estimate a scaling factor for each full-journey *itinerary*, which is defined in this context as a unique sequence of *fare-transaction nodes* observed to have been visited by at least one passenger. The concept of transaction nodes and itineraries can be illustrated using the hypothetical transit network shown in Figure 5-1 (referred to hereafter as the *test network*).

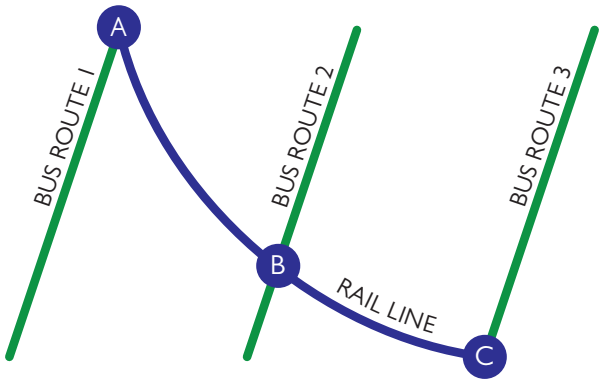


Figure 5-1. The test transit network.

In the context of full-journey expansion, a transaction node is defined as a node on the network at which a rider can tap an Oyster card.

1 A network is considered closed if each passenger has one entry and one exit transaction but no interchange transactions. The London rail network can be considered closed because transfers are made behind the gate, yielding a single journey stage per system entry and exit.

Each transaction node is uniquely identified by either a station and movement (entry vs. exit) or a route and direction (inbound vs. outbound). For example, the test network consists of the following twelve transaction nodes (inbound is considered northbound in Figure 5-1):

| | |
|---|---|
| Station A entry | Station A exit |
| Station B entry | Station B exit |
| Station C entry | Station C exit |
| Route 1 inbound | Route 1 outbound |
| Route 2 inbound | Route 2 outbound |
| Route 3 inbound | Route 3 outbound |

This simple network yields 38 possible combinations of transaction nodes that can be traversed on any passenger journey. For example, a common itinerary might be station A entry to station C exit, while a less common itinerary might be route 1 inbound to station A entry to station B exit to route 2 inbound. It follows that every bus journey stage relates to a single transaction node (identified by the route and direction), while each rail journey stage consists of two nodes (an entry station and an exit station).

While this network of three routes and three stations yields a manageable 38 potential itineraries, the Oyster network contains over 1,300 station codes and over 800 routes. A very large number of potential itineraries can be defined, many entailing hundreds of interchanges and therefore being extremely unlikely to be used by any passengers. It is for this reason that the set of itineraries included in the analysis only includes those sequences of transaction nodes that have been observed to have been taken by at least one passenger on the day being analyzed.

The estimation of full intermodal passenger-journey flows therefore requires the estimation of an expansion factor for each full-journey itinerary observed in the sample. The approach taken is conceptually related to some of the methods used to estimate traditional OD matrices, and is presented after a review of the work upon which it builds.

### 5.2.1 *Iterative Proportional Fitting on Closed Networks*

The problem of expanding, or scaling, full passenger journeys is in some ways similar to that of expanding OD flows on a single bus route or closed rail network. In these cases a sample of passenger flows can be obtained or inferred, and the flows can be scaled to match a set of control totals measured at each start and end point (or in the case of full journeys, at each interchange point as well). The iterative proportional fitting (IPF) method (Deming and Stephan 1940) is often used to solve the OD matrix expansion problem for rail networks and bus routes (Ben Akiva et al 1985, Wilson et al 2008, McCord et al 2010), and has also been applied to the London Underground network (Gordillo 2006, Chan 2007).

If a rail origin–destination matrix is viewed as a contingency table, with each row representing an origin and each column representing a destination, the IPF method attempts to estimate a scaling factor for each row (origin) and another for each column (destination) that satisfy the following system of equations:

$$T_{o,d} = \mathrm{t}_{o,d} \cdot \alpha_o \cdot \beta_d \qquad \forall\, o,d \tag{5.1}$$

$$\sum_{d \in \mathrm{D}} T_{o,d} = \mathrm{C}_o \qquad \forall\, o \tag{5.2}$$

$$\sum_{o \in \mathrm{O}} T_{o,d} = \mathrm{C}_d \qquad \forall\, d \tag{5.3}$$

where

- $T_{o,d}$ is the estimated total passenger flow from origin station $o$ to destination station $d$,
- $\mathrm{t}_{o,d}$ is the observed sample passengers flow from origin station $o$ to destination station $d$,
- $\alpha_o$ is the estimated scaling factor for origin station $o$,
- $\beta_d$ is the estimated scaling factor for destination station $d$,
- $\mathrm{C}_o$ is the control total for entries at origin station $o$,
- $\mathrm{C}_d$ is the control total for exits at destination station $d$.

In other words, scaling factors must be chosen for each origin and each destination such that the sample OD flows (collectively called the *seed matrix*) are scaled to satisfy each station's entry and exit control totals.

In this way IPF yields a scaling factor for each OD pair, but each of these scaling factors is the product of two other scaling factors: the entry and exit factors $\alpha_o$ and $\beta_d$. When applied to the scaling of OD matrices from Oyster data, this relationship necessitates the assumption that for each OD pair, the proportion of travel successfully recorded with Oyster is a function of the station of entry and the station of exit. This relationship also provides the mechanism by which the IPF method solves the system of equations.

Figure 5-2 depicts the IPF problem as a contingency table, using the rail subsystem of the test network (stations A, B, and C) as an example. Each row corresponds to an origin station; each column, a destination station; and each cell, an OD pair. Each sample flow ($t_{o,d}$) is multiplied by its corresponding scaling factors ($\alpha_o$ and $\beta_d$), and the products are summed for each row ($\sum_o$) and each column ($\sum_d$). Equations 5.1 through 5.3 show that scaling factors must be chosen such that each row sum equals its corresponding origin control total and each column sum equals its corresponding destination control total.

A solution is obtained by first setting all scaling factors to 1, then repeatedly solving two of the three equations, alternating between solving for row factors (equations 5.1 and 5.2) and column factors (equations 5.1 and 5.3). In the example, this is achieved by setting all $\alpha$ to the cor-

| | $\beta_A$ | $\beta_B$ | $\beta_C$ | | |
|---|---|---|---|---|---|
| $\alpha_A$ | 0 | $t_{A,B}$ | $t_{A,C}$ | $\sum_{oA}$ | $C_{oA}$ |
| $\alpha_B$ | $t_{B,A}$ | 0 | $t_{B,C}$ | $\sum_{oB}$ | $C_{oB}$ |
| $\alpha_C$ | $t_{C,A}$ | $t_{C,B}$ | 0 | $\sum_{oC}$ | $C_{oC}$ |
| | $\sum_{dA}$ | $\sum_{dB}$ | $\sum_{dC}$ | | |
| | $C_{dA}$ | $C_{dB}$ | $C_{dC}$ | | |

Figure 5-2. IPF contingency table for the test network shown in Figure 5-1.

responding quotient $C_o / \sum_o$, and then setting all $\beta$ to the corresponding quotient $C_d / \sum_d$. Since the total estimated flow for each OD pair, $T_{o,d}$, is a function of two scaling factors—one in the entry dimension and another in the exit dimension—the adjustment of factors in one dimension can satisfy that dimension's control totals but typically upsets the balance in the other dimension. With successive iterations, however, the differences between the control totals and marginal totals (the row and column sums) often converge toward zero.

Iterative proportional fitting, when convergent, yields the maximum-likelihood estimate for each OD pair's scaling factor ($\alpha_o \beta_d$) (Halberman 1974). Ben Akiva et al (1985) compared IPF to other matrix-expansion methods, including constrained generalized least squares (Björck 1996), constrained maximum-likelihood, and an intervening opportunity model. They found IPF to be preferable to the other methods due to its "computational ease without loss of accuracy."

Ben Akiva (1987) also noted that IPF will converge to a unique solution (a *biproportional fit*) if the seed matrix contains no zeros. If zeros do exist, either because of infeasible combinations of origins and destinations (*structural zeros*, such as those OD pairs in the example that start and end at the same station) or because the flow between a feasible OD pair was not included in the sample (*sampling zeros*), the estimated total flow for that OD pair will remain at zero. It follows that there will be no direct relationship between that OD pair's origin and destination scaling factors (although they might influence each other indirectly through other OD pairs), and that successive cycles might not converge. Pukelsheim (2012) shows that in the case of non convergence, successive iterations tend to oscillate between two accumulation points, one of which is approached during the origin phase of the adjustment cycle; the other, during the destination phase.

### 5.2.2 *Applications of IPF on London's Public Transport Network*

Gordillo (2006) used IPF to build an OD matrix for the London Underground, using Oyster data to construct the seed matrix and station gateline data to derive the control totals. To correct for the bias of some non-gated stations being undercounted, a method was devised which supplements data from non-gated stations with data from manual counts and from the RODS travel survey.

Gordillo's OD matrix describes the London Underground's travel activity for a full weekday, but Chan (2007) builds upon this work by devising a method for constructing a separate matrix for each of TfL's six time periods, as listed in Table 5-1:[2]

Table 5-1. London Underground weekday time periods

| Time Period | Hours |
|---|---|
| Early morning | 5:30–6:59 AM |
| AM peak | 7:00–9:59 AM |
| Midday | 10:00 AM–3:59 PM |
| PM peak | 4:00–6:59 PM |
| Evening | 7:00–9:59 PM |
| Late evening | 10:00 PM–12:30 AM |

If each time-period OD matrix were to contain only the passenger trips that started and ended during the specified time period, passenger trips that began during one time period and ended during another would be excluded from the set of matrices. Chan therefore assigns trips to time periods based on the time of the passenger's entry tap, regardless of when the rider tapped out of the system.[3] The sample (the seed matrix) will then include some portion of passenger trips that end during later time periods, and which therefore contribute to the later periods' control totals rather than those of the entry time period. Chan's solution is to adjust the stations' exit control totals to incorporate some portion of the current period's total and some portion of the subsequent period's total. This exit-adjustment methodology can be generalized to the adjustment of rail-exit, rail-entry, or bus-boarding totals (all of which are required for

2 For a discussion of the homogeneity within time periods see Ji et al (2011).

3 Alternatively, trips could be assigned based on their end times, regardless of when the trip started. This second approach might be more useful when analyzing the AM peak, when commuters tend to choose various start times based on their desired (and often similar) arrival times. Conversely, the former approach might be more useful for the PM peak, when passengers tend to enter the system at somewhat similar times (such as after work) but whose exit times vary based on their travel times. For consistency, however it is conventional to choose one of the two approaches for all time periods.

the scaling of full-journey matrices[4]), spanning any number of arbitrarily sized time periods, as follows:

$$r_{n,p,s} = \frac{\mathrm{t}_{n,p,s}}{\sum_{i \in \mathrm{P}} \mathrm{t}_{n,p,i}} \qquad \forall\, n \in \mathrm{N},\, p \in \mathrm{P},\, s \in \mathrm{P} \tag{5.4}$$

$$\hat{\mathrm{C}}_{n,\mathrm{A}} = \sum_{p \in \mathrm{A}} \sum_{s \in \mathrm{P}} r_{n,p+s,s} \cdot \mathrm{C}_{n,p+s} \qquad \forall\, n \in \mathrm{N} \tag{5.5}$$

where

- N is the set of transaction nodes (such as rail entry or exit stations) on the network,
- P is the set of recording periods: time periods for which control-total data are collected for a given service day,
- A is the analysis period: the time range for which the matrix is being constructed, which consists of a subset of the recording periods contained in P,
- $s$ is the span, or number of recording periods, between a given fare transaction and the first transaction of the corresponding journey,
- $\mathrm{t}_{n,p,s}$ is the number of passengers in the seed matrix who tapped at node $n$ during recording period $p$, and whose journeys began $s$ recording periods before $p$,
- $r_{n,p,s}$ is the ratio (or proportion), of all passengers in the seed matrix who tapped at node $n$ during recording period $p$, whose journeys began $s$ recording periods earlier,
- $\mathrm{C}_{n,p}$ is the unadjusted recording-period control total: the number of passengers counted at node $n$ during recording period $p$,
- $\hat{\mathrm{C}}_{n,\mathrm{A}}$ is the adjusted analysis-period control total: the estimated number of passengers who passed through node $n$, regardless of recording period, whose journeys began during analysis period A.

4 Since Chan's OD matrices are defined by the time period of the entry transaction, only the exit transaction count must be adjusted (the time period of the entry tap is by definition the time period of the passenger trip). But since any full journey can include multiple station entries, station exits, and bus boardings, all control totals must be adjustable to reflect the time period of the first transaction in the passenger's journey (for example, a station entry might be part of an interchange that occurs in a later time period than the first transaction in that journey).

Equation 5.4 illustrates that the timespan proportions ($r_{n,p,s}$) are derived entirely from the seed matrix. It is assumed that the timespan proportions in the sample are a reasonable proxy for those of the population, which are unknown. Under this assumption, equation 5.5 applies the proportions to the observed control totals, yielding the adjusted control totals.

Figure 5-3 illustrates the control-total adjustment process, using the example of the exit counts at a London Underground station, with each recording period corresponding to an hour. The table at the left contains the timespan proportions, inferred from the Oyster sample. These are multiplied by the unadjusted hourly totals, distributing them among the

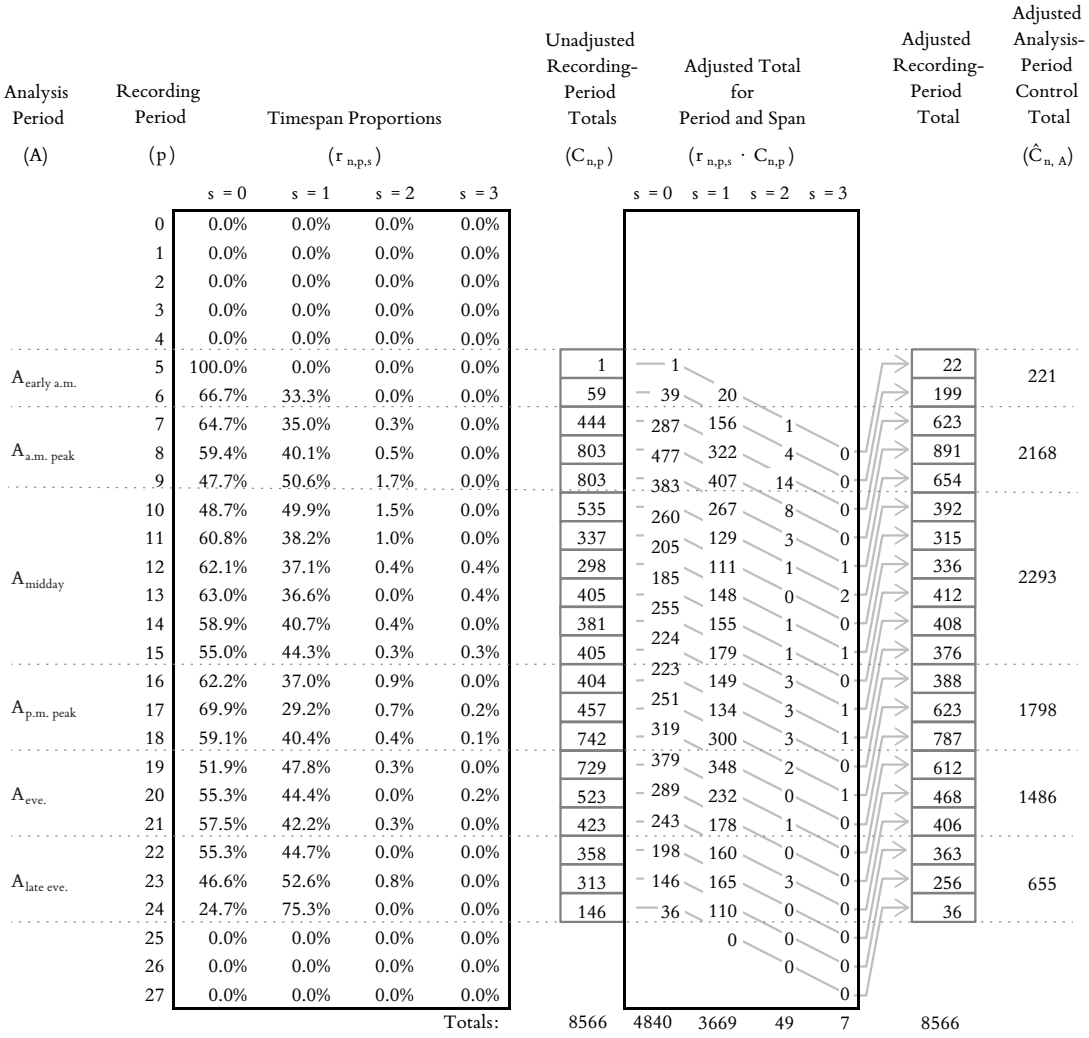


Figure 5-3. An example of the control-total adjustment process, using exit counts from a London Underground station.

four observed spans, as shown in the table at the right. The components of each span/period count are then summed diagonally to yield the adjusted hourly totals, which are in turn aggregated by analysis time period. The diagonal summation in the figure is reflected in equation 5.5, where the hour dimension is incremented by $p + s$.

Chan also adapted the IPF procedure to ensure that all scaling factors for each OD pair were no less than 1.0, since Oyster transactions are a subset of a station's fare activity and therefore cannot be greater than the station's control total. After adjusting the control totals, Chan constructs OD matrices by applying Gordillo's method to each time period, substituting $\hat{C}_{n,p}$ for $C_n$ in equations 5.2 and 5.3 and adding the constraints $\alpha_o \geq 1$ and $\beta_d \geq 1$. (A more robust method for applying these constraints to IPF is presented in sections 5.4.1 and 5.5.1.)

## 5.3 INPUT DATA

Like the IPF method, the full-journey expansion process requires a set of sample data and control totals, as follows:

*Origin-, destination-, and interchange-inferred Oyster data*. After origins, destinations, and interchanges (ODX) have been inferred for Oyster transactions using the methods described in chapters 3 and 4, the Oyster records can be used to construct a seed matrix of full passenger journeys, which can in turn be scaled to match a set of control totals. The ODX-inferred Oyster data only include those journey stages that were successfully processed by the ODX-inference tools: any data that were discarded during any of the three inference processes will not be represented in the seed data and must be accounted for by the scaling process. Additionally, any Oyster trips that were only partially recorded, or any travel that was made without an Oyster card, will not have been included in the seed matrix. Because the seed matrix is an approximately 75 percent sample of TfL travel, the probability of it containing a significant number of sampling zeros is very low.

*Rail Transaction Totals*. Transaction counts from electronic station gates, or *gatelines*, are available for most gated stations that accept Oyster cards.

Gateline counts are aggregated by date, hour, station, movement (either entry or exit), and payment type. These totals include transactions made with the gates' Oyster readers as well as the magnetic-stripe readers used to process paper tickets. There are some stations at which transaction totals are expected to undercount the number of actual transactions, such as ungated stations at which passengers using valid travel passes are not required to tap. Additionally, at the time of this writing, counts for some stations (such as some of the new stations on the East London Line) are not yet available.

*Bus Farebox Counts*. Data from bus fareboxes, or electronic ticket machines (ETMs), include counts of each Oyster transaction, as well as of many non-Oyster transactions such as paper tickets and visually inspected passes. Bus operators are required to record the use of visually inspected passes by pressing a button on the ETM but adherence to this rule varies among drivers. Because of this variation, and because of the current difficulty in obtaining disaggregate ETM data (as described in section 3.2.1), bus control totals are obtained at the route level rather than at the stop level, with the route direction being recorded as well. While stop-level counts may be useful for route-level analyses, the use of route-level control totals provides a reasonable level of aggregation for the scaling of intermodal and system-wide travel activity: rider activity can be analyzed at the stop level, but all stops will contribute to the ETM total of their route and direction. If stop-level data can be more easily obtained in the future, however, the scaling processes described in this chapter can still be applied: it will simply process a larger number of ODX combinations.

### 5.4.1 *Problem Definition Revisited*

The problem of estimating expansion factors for full-journey itineraries is illustrated in Figure 5-4. The entry gates at Station A and the exit gates at Station B (from the test network in Figure 5-1) have different control totals ($\hat{C}_{Ao}$ and $\hat{C}_{Bd}$, estimated using the process generalized from Chan in section 5.2.2), and the flows through these nodes comprise various full-journey itineraries. The observed flow of each itinerary ($t_1$ through $t_{37}$) must be scaled by some factor ($\alpha_1$ through $\alpha_{37}$) in order to estimate the additional flow that was not observed or inferred from the Oyster data. This additional flow, $\Delta_{Ao}$ and $\Delta_{Bd}$, must be allocated to the itineraries that constitute each node's flow. In this example, itinerary scaling factors must be chosen that satisfy the following two equations:

$$\Delta_{Ao} = t_1\alpha_1 + t_2\alpha_2 + t_3\alpha_3 + t_4\alpha_4 + t_5\alpha_5 + t_{17}\alpha_{17} + t_{18}\alpha_{18} + t_{19}\alpha_{19} + t_{20}\alpha_{20} + t_{21}\alpha_{21}$$
$$\Delta_{Bd} = t_1\alpha_1 + t_3\alpha_3 + t_{11}\alpha_{11} + t_{12}\alpha_{12} + t_{17}\alpha_{17} + t_1\alpha_{19} + t_{32}\alpha_{32} + t_{33}\alpha_{33} + t_{36}\alpha_{36} + t_{37}\alpha_{37}$$

There are multiple solutions to each node's equation. For example, all of the unobserved flow entering Station A could be allocated to itinerary 1 by setting $\alpha_1$ to $\Delta_{Ao}/t_1$ while setting all other scaling factors related to the node ($\alpha_1$ through $\alpha_5$ and $\alpha_{17}$ through $\alpha_3$) to zero. The unobserved flow could similarly be allocated exclusively to any of the other itineraries that contribute to Station A's entry count. At the network level, however, the problem is constrained by the relationships between transaction nodes.

Transaction nodes are related to each other through itineraries. In Figure 5-4, for example, itineraries 1, 3, 17, and 19 are common to both nodes. Itinerary 1 starts at Station A and terminates at Station B, while itinerary 3 starts at Station A and then interchanges from Station B to Route 1 (see figures 5-1 and 5-2). Most of the itineraries that constitute Station A's entry flow do not pass through Station B's exit gates, but contribute to the totals of other transaction nodes not shown in the figure. Since each itinerary is given its own scaling factor, the factors for itineraries 1, 3, 17, and 19 contribute to the unobserved flow of both nodes in the illustration ($\Delta_{Ao}$ and $\Delta_{Bd}$). This relationship is illustrated for itinerary 1, for which the

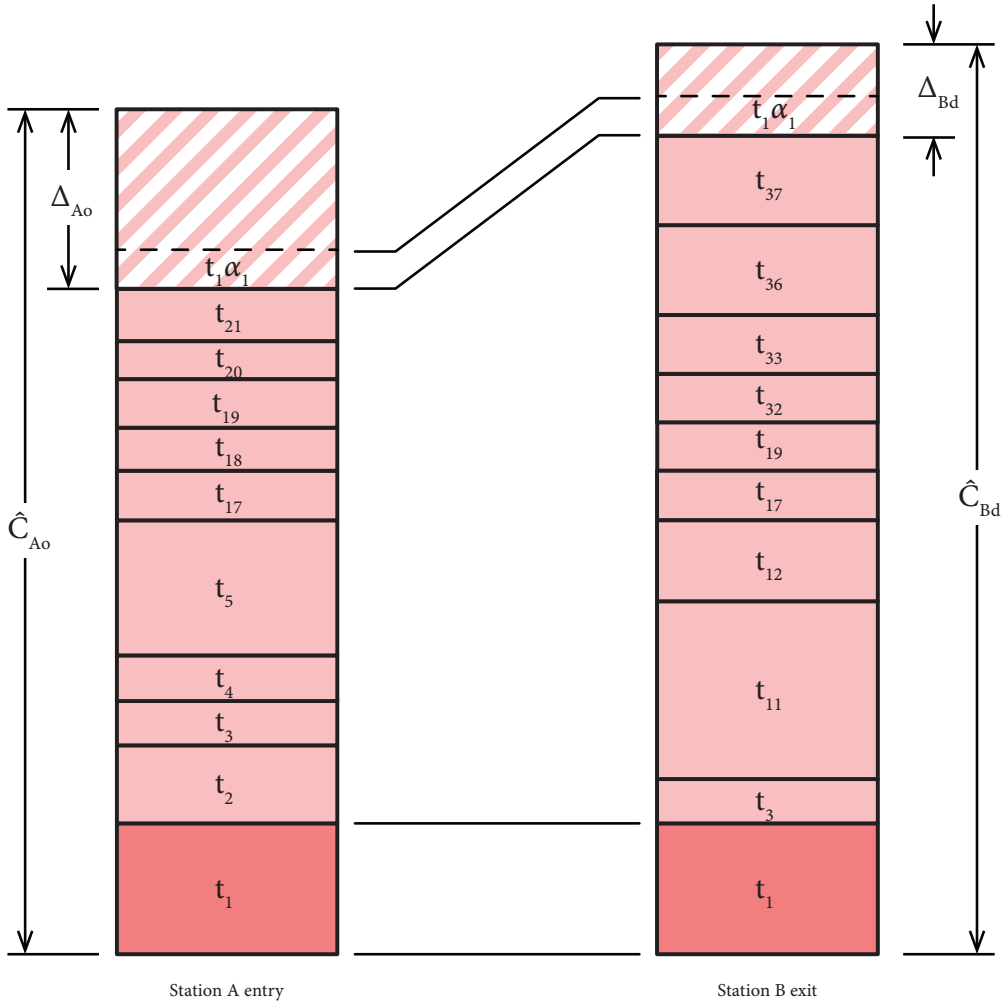


Figure 5-4. Example of estimated passenger flow on the test network.

sample flow ($t_1$) is shown in a darker shade and the estimated non-sample flow ($t_1\alpha_1$) is delimited by a dashed line.

While the relationship between full-journey itineraries and transaction-node control totals constrains the problem, there is no guarantee that a solution to the problem will be unique. An ideal scaling method would therefore derive the most likely of all solutions to the problem, much like IPF attempts to find the most likely solution to the OD-matrix-expansion problem. The method proposed in this section therefore builds upon the IPF approach, but modifies it to reflect the relationship between transaction nodes and full-journey itineraries.

### 5.4.2 *Problem Formulation and Solution*

The relationship between full-journey itineraries and transaction-node control totals can be formulated as follows:

$$\Delta_n = \hat{C}_n - \sum_{i \in I} t_i \cdot \mathbf{B}[n, i] \qquad \forall\, n \in N \tag{5.6}$$

$$\sum_{i \in I} t_i \cdot \alpha_i \cdot \mathbf{B}[n, i] = \Delta_n \qquad \forall\, n \in N \tag{5.7}$$

$$T_i = t_i \cdot (1 + \alpha_i) \qquad \forall\, i \in I \tag{5.8}$$

where

- N is the set of all Oyster transaction nodes (bus routes inbound, bus routes outbound, stations of entry, and stations of exit),
- I is the set of all full-journey itineraries (unique sequences of Oyster transaction nodes inferred to have been taken by at least one cardholder),
- $\Delta_n$ is the difference between the control total and the sample total at transaction node $n$,
- $\hat{C}_n$ is the adjusted control total for transaction node $n$, as defined in equations 5.4 and 5.5 (it is assumed that all variables correspond to a single analysis period, $A$),
- $t_i$ is the flow of cardholders inferred in the seed matrix to have taken itinerary $i$,
- $\mathbf{B}[n, i]$ is an incidence matrix—a binary matrix of ones and zeros—indicating whether node $n$ is traversed by itinerary $i$,
- $\alpha_i$ is the amount by which the sample flow for itinerary $i$ will be scaled,
- $T_i$ is the estimated total flow of passengers on itinerary $i$.

The purpose of equation 5.6 is to ensure that no itineraries are scaled downward. Chan (2007) achieved a similar goal by imposing the constraint $\alpha_o\beta_d \geq 1$ in her IPF methodology (see section 5.2.2), which in the full-journey expansion problem would correspond to a constraint of $\alpha_i \geq 1$. Applying such a constraint in an iteratively solved problem, however, can inhibit convergence during any one phase of the cycle by forbidding

the selection of some factors that would otherwise (temporarily) satisfy the control total, thereby propagating error to a subsequent phase.

By scaling the seed itinerary flows downward to the difference, $\Delta_n$, rather than upward to the entire control total, $\hat{C}_n$, the constraint $\alpha_i \geq 1$ becomes unnecessary, since scaling factors now need only be greater than or equal to zero (rather than one). The corresponding constraint $\alpha_i \geq 0$ need not be applied because non-negativity is guaranteed by the definition of a scaling factor ($\alpha_i$) as the quotient of two non-negative numbers (the control total and the sample total). After scaling factors are calculated in this way, they are added back to the seed itinerary flow to derive the total itinerary flow, as shown in equation 5.8.

Since the unobserved flows are unknown, the goal of the full-journey expansion process is to find the most uniform scaling factors that satisfy equation 5.7. If we assume that the best estimate is that all itineraries are scaled as evenly as possible, we can seek to minimize the variance among all scaling factors. This could be approached as a least-squares optimization problem, where the objective function—the scaling factor variance—is minimized, subject to the satisfaction of equation 5.7. But with over 4,000 transaction nodes in the Oyster network and roughly 800,000 different itineraries observed on a typical weekday,[5] the optimization problem requires a matrix of approximately 3.2 billion elements (the product of the two values), making it too large to process on a typical computer.

A somewhat similar problem is solved by IPF, at least in the case of traditional OD matrices. The convergence of errors between sample and control totals is the direct result of repeatedly adjusting each scaling factor, which effectively distributes the potential error among all scaling factors, leading in many cases to the attainment of a maximum-likelihood estimate (Halberman 1974). Since this meets the goals of achieving likely estimates while satisfying control totals, a similar approach is taken when solving the full-journey scaling problem.

In each phase of the IPF cycle, a scaling factor is estimated for each node by dividing the desired flow (the control total) by the current flow (the sample flow scaled by the expansion factors of the previous iteration). A less direct approach must be taken for full journeys, since expansion factors exist only in the itinerary dimension and control totals exist only

5 800,000 itineraries are typically observed when aggregating itineraries to the bus-route level. When further disaggregated by bus stop, there are rougly 1.7 million daily itineraries.

in the transaction-node dimension (in IPF, factors and nodes exist in both the origin dimension and the destination dimension).

Returning to the example in Figure 5-4, a scaling factor, $\alpha_1$, must be estimated for the observed itinerary flow $t_1$. If all nodes on the network are ignored except node A, we might assume that the distribution of itineraries in the node's unobserved flow is the same as the distribution of itineraries in the sample flow—that is, we would set $\alpha_1$ equal to $\Delta_{Ao}/(\hat{C}_{Ao} - \Delta_{Ao})$. But it is clear from the example that the proportion of a node's sample flow assigned to an itinerary is not necessarily equal to the proportion of the unobserved flow assigned to that itinerary. In this case, $t_1$ constitutes a greater proportion of the sample flow through node A than through node B, while $t_1\alpha_1$ constitutes a greater proportion of the unobserved flow through node B than through node A. The approach taken is therefore to choose an itinerary scaling factor that is the average of the ratios of the itinerary's seed flow to the seed flows of its constituent nodes. This problem is solved iteratively, and can be formulated as:

$$M_n = \sum_{i \in I} t_i \cdot \alpha_i \cdot \mathbf{B}[n, i] \qquad \forall\, n \in N \tag{5.9}$$

$$\alpha_i = \alpha_i^{\star} \frac{\sum_{n \in i} \frac{\Delta_n}{M_n}}{\sum_{n \in i} 1} \qquad \forall\, i \in I \tag{5.10}$$

where

- $M_n$ is the marginal transaction-node total: the non-Oyster flow through node $n$, as calculated using the currently selected scaling factor,
- $\alpha_i^{\star}$ is the itinerary scaling factor chosen during the previous iteration of the problem,

  and all other symbols are as defined earlier in this chapter.

Each $\Delta_n$ is calculated once, as shown in equation 5.6. All $\alpha_i$ are then set to 1.0, and equations 5.9 and 5.10 are repeatedly solved until the marginal totals converge upon the set of $\Delta_n$.

Section 5.2.1 showed that convergence is reached in IPF by alternating between the adjusting of entries and exits. The adjustment of any entry

scaling factor can directly affect any number of exit totals, but cannot affect any other entry totals, and vice versa. All entries (rows) can therefore be adjusted simultaneously or in any order, and the process is then repeated for all exits (columns). Since the full-journey problem contains scaling factors only in the itinerary dimension and control totals only in the node dimension, the adjustment of any itinerary scaling factor can affect the choice of scaling factors for other itineraries.

Because of this interaction between itinerary scaling factors, the order in which factors are estimated can affect the results. If the scaling factors in the example were calculated in order, itinerary 1 would be allocated without being constrained by any other itinerary's scaling factor, then itinerary 3 would be allocated to some portion of $\Delta_A$ not already allocated to itinerary 1. Since there are approximately 4,000 transaction nodes on the Oyster network but more than 800,000 itineraries, it is possible that some number of control totals will be satisfied by the scaling of some subset of their constituent itineraries before all of their itineraries are processed. This bias is eliminated by calculating scaling factors as described in equations 5.9 and 5.10, but not applying the factors until the end of each iteration, after all itineraries' scaling factors have been calculated.

## 5.5 EXAMPLE: TEST NETWORK

The methodology described in the previous section was first implemented in a spreadsheet program and applied to the test network (Figure 5-1). Unlike a real transport network, the total itinerary flows (which are typically unknown) can be defined in advance, and the method's outputs can be validated against them. A rail-only OD matrix is then derived from the full-journey matrix, and the results are tested against a traditional OD matrix generated using IPF.

### 5.5.1 *Validation*

The full-journey scaling algorithm was applied to the test network, as illustrated in Figure 5-5. The actual itinerary counts and Oyster penetration rates (the share of each itinerary count paid for or validated using Oyster) are unknown in a real network, but were chosen in this example.

The actual itinerary flows are multiplied by the Oyster penetration rates to derive the itinerary seed flows, which along with the control totals are the independent variables in the algorithm.

Below the actual and seed itinerary counts in the figure are the estimated itinerary scaling factors, which are the only adjustable parameters in the problem: the algorithm solves for these values. Beneath the scaling factors is the incidence matrix (**B** in equations 5.6 and 5.7), which defines the relationships between itineraries and transaction nodes by indicating which of the twelve transaction nodes constitute each of the 37 observed itineraries (there are 38 possible itineraries, but it is assumed that not all were used by riders during the analysis period).

Comparing the full-journey scaling problem illustrated in Figure 5-5 to the IPF contingency table in Figure 5-2, it can be seen that IPF has control totals and scaling factors in both dimensions (the origin, or vertical dimension, and the destination, or horizontal dimension). Since the full-journey scaling problem contains flows in two or more dimensions (an origin, a destination, and any number of intermediate interchange nodes),

| Itinerary ($i$) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Actual itinerary counts ($T$) | 200 | 176 | 22 | 29 | 38 | 84 | 27 | 123 | 28 | 16 | 87 | 16 | 134 | 21 | 74 | 58 | 16 | 21 |
| Oyster penetration rate | 50% | 68% | 91% | 86% | 79% | 95% | 67% | 73% | 82% | 75% | 86% | 94% | 82% | 95% | 81% | 86% | 75% | 90% |
| Itinerary seed counts ($t$) | 100 | 120 | 20 | 25 | 30 | 80 | 18 | 90 | 23 | 12 | 75 | 15 | 110 | 20 | 60 | 50 | 12 | 19 |
| | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| Itinerary scaling factors ($\alpha$) | 0.68 | 0.46 | 0.59 | 0.39 | 0.43 | 0.26 | 0.29 | 0.26 | 0.22 | 0.26 | 0.36 | 0.31 | 0.16 | 0.19 | 0.07 | 0.39 | 0.59 | 0.41 |
| | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| Station A entry: | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Station B entry: | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Station C entry: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Station A exit: | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Station B exit: | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| Station C exit: | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Bus 1 inbound | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| Bus 1 outboond | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| Bus 2 inbound | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus 2 outbound | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus 3 inbound | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bus 3 outbound | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figure 5-5. Formulation and results of the full-journey scaling process, as applied to the test network.

all nodes are collapsed to a single dimension, with all three node types (station entries, station exits, and bus boardings) listed on the vertical axis. Since the itineraries correspond to the horizontal axis, scaling factors exist only in the horizontal dimension and control totals exist only in the vertical. As with IPF, the scaling factors are initially set to zero. The problem is then iteratively solved as described in section 5.4.2.

The model was applied using various parameter settings in order to test its robustness. If the iterative averaging process scales the itineraries relatively evenly (within the constraints imposed by the control totals), the results should be more accurate when there is less variance between the itineraries' penetration rates. Figure 5-6 illustrates the convergence of the marginal totals (the estimated unobserved flow on each node) towards the nodes' control totals over 20 iterations of the algorithm. The first data set, which contained Oyster penetration rates with a standard deviation of 6 percent, converged to match all control totals within eight iterations. The second data set, which had a standard deviation of 20 percent, con-

| 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | 13 | 5 | 60 | 64 | 19 | 6 | 23 | 3 | 84 | 71 | 36 | 32 | 24 | 13 | 41 | 49 | 24 | 13 |
| 86% | 69% | 80% | 83% | 70% | 84% | 67% | 70% | 67% | 83% | 92% | 97% | 78% | 75% | 92% | 73% | 71% | 67% | 69% |
| 6 | 9 | 4 | 50 | 45 | 16 | 4 | 16 | 2 | 70 | 65 | 35 | 25 | 18 | 12 | 30 | 35 | 16 | 9 |
| * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |
| 0.52 | 0.39 | 0.35 | 0.26 | 0.08 | 0.26 | 0.29 | 0.26 | 0.22 | 0.08 | 0.28 | 0.14 | 0.17 | 0.30 | 0.27 | 0.19 | 0.21 | 0.34 | 0.30 |
| * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * | * |

| | | | | | | | | | | | | | | | | | | | | (*M*) Marginal | | (Ĉ) Control |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Σ → | 181 | vs. | 182 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Σ → | 67 | vs. | 68 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Σ → | 92 | vs. | 90 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | Σ → | 75 | vs. | 76 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | Σ → | 138 | vs. | 139 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Σ → | 127 | vs. | 125 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Σ → | 27 | vs. | 26 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | Σ → | 41 | vs. | 41 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Σ → | 23 | vs. | 23 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | Σ → | 29 | vs. | 28 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Σ → | 40 | vs. | 40 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | Σ → | 59 | vs. | 59 |

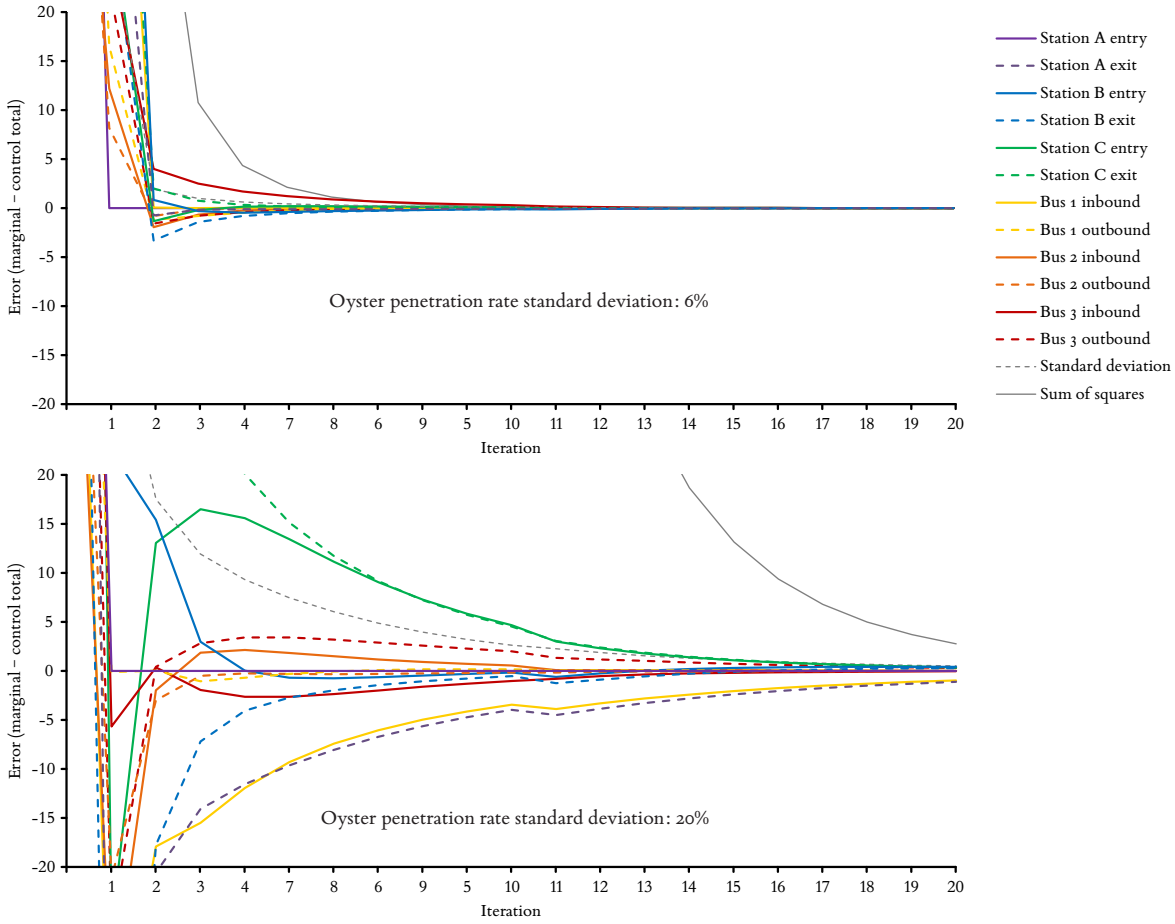


Figure 5-6. Convergence of scaled transaction node flows toward control totals.

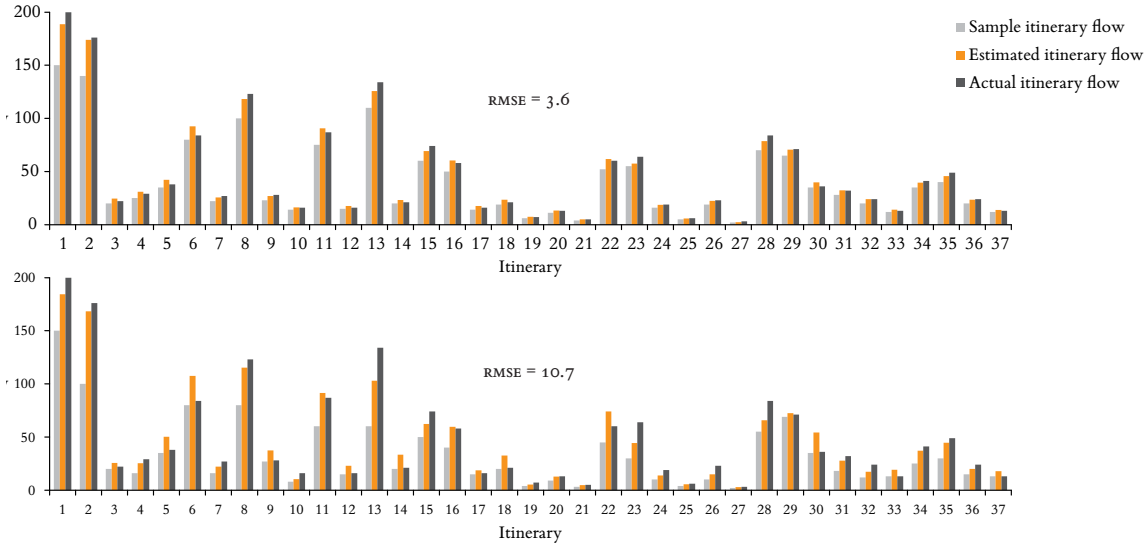


Figure 5-7. Comparison of actual and estimated itinerary flows.

verged toward all control totals with the exception of two, which each had errors of one rider.

Figure 5-7 compares the estimated itinerary flows to the actual flows (which are unknown in a real network) and to the sample flows (which are independent variables in the algorithm). On the data set with less variability, 34 of the 37 estimated flows provided closer estimates of the actual flows than did the sample flows, and the root-mean-square error (RMSE) between the actual and estimated totals was 3.6. The model with greater deviation provided an equal or better estimate on 29 of the 37 itineraries, with an RMSE of 10.7.

### 5.5.2 *Comparison to Iterative Proportional Fitting*

While the full-journey matrix-expansion process estimates scaling factors for full-journey itineraries, these itineraries comprise individual journey stages that can be compared to traditional single-mode OD matrices. For example, the flow from Station A to Station B in the test network (see Figure 5-4) can be derived by adding the scaled flows of all itineraries that include that OD pair, or in this case:

$$\text{Flow from A to B} = t_1(1 + \alpha_1) + t_3(1 + \alpha_3) + t_{17}(1 + \alpha_{17}) + t_{19}(1 + \alpha_{19})$$

By calculating flows for all OD pairs of the rail network in this way, a rail-only OD matrix can be derived from the full-journey matrix.

The resultant rail OD matrix was compared to another rail OD matrix, generated using IPF on the same data. The IPF algorithm was implemented as described by Chan (2007) and Gordillo (2006), which were reviewed in section 5.2. The only modification to the algorithm was to scale the OD flows to the unallocated portion of the control totals rather than to the entire control totals, as described in the full-journey estimation process (section 5.4.2). The modified IPF algorithm is formulated as follows:

$$\Delta_o = C_o - \sum_{d \in D} t_{o,d} \qquad \forall\, o \qquad (5.11)$$

$$\Delta_d = C_d - \sum_{o \in O} t_{o,d} \qquad \forall\, d \qquad (5.12)$$

$$\sum_{d \in D} t_{o,d} \cdot \alpha_o \cdot \beta_d = \Delta_o \qquad \forall\, o \qquad (5.13)$$

$$\sum_{o \in O} t_{o,d} \cdot \alpha_o \cdot \beta_d = \Delta_d \qquad \forall\, d \qquad (5.14)$$

$$T_{o,d} = t_{o,d} \cdot (1 + \alpha_o \cdot \beta_d) \qquad \forall\, o, d \qquad (5.15)$$

where

$\Delta_o$ is the difference between station $o$'s control-total entries and sampled entries,

$\Delta_d$ is the difference between station $d$'s control-total exits and sampled exits,

and all other notation is as defined for equations 5.1 through 5.5.

Since the full-journey matrix expansion algorithm is more constrained than the IPF method (both are constrained by the matching of node control totals while full-journey expansion adheres to the additional constraint of having to relate transaction nodes through itineraries), it should be expected that there is more variation between the OD scaling factors derived from the full-journey matrix than between the OD scaling factors calculated through IPF. The small number of OD pairs on the test network yields too small a sample to test this difference in variation (this is tested in section 5.6.2 using the London network), but the comparison of each OD pair's scaling factors in Figure 5-8 shows that both approaches yield similar scaling factors, with a greater variation in Oyster penetration rates leading to greater variation between the results of the two algorithms.

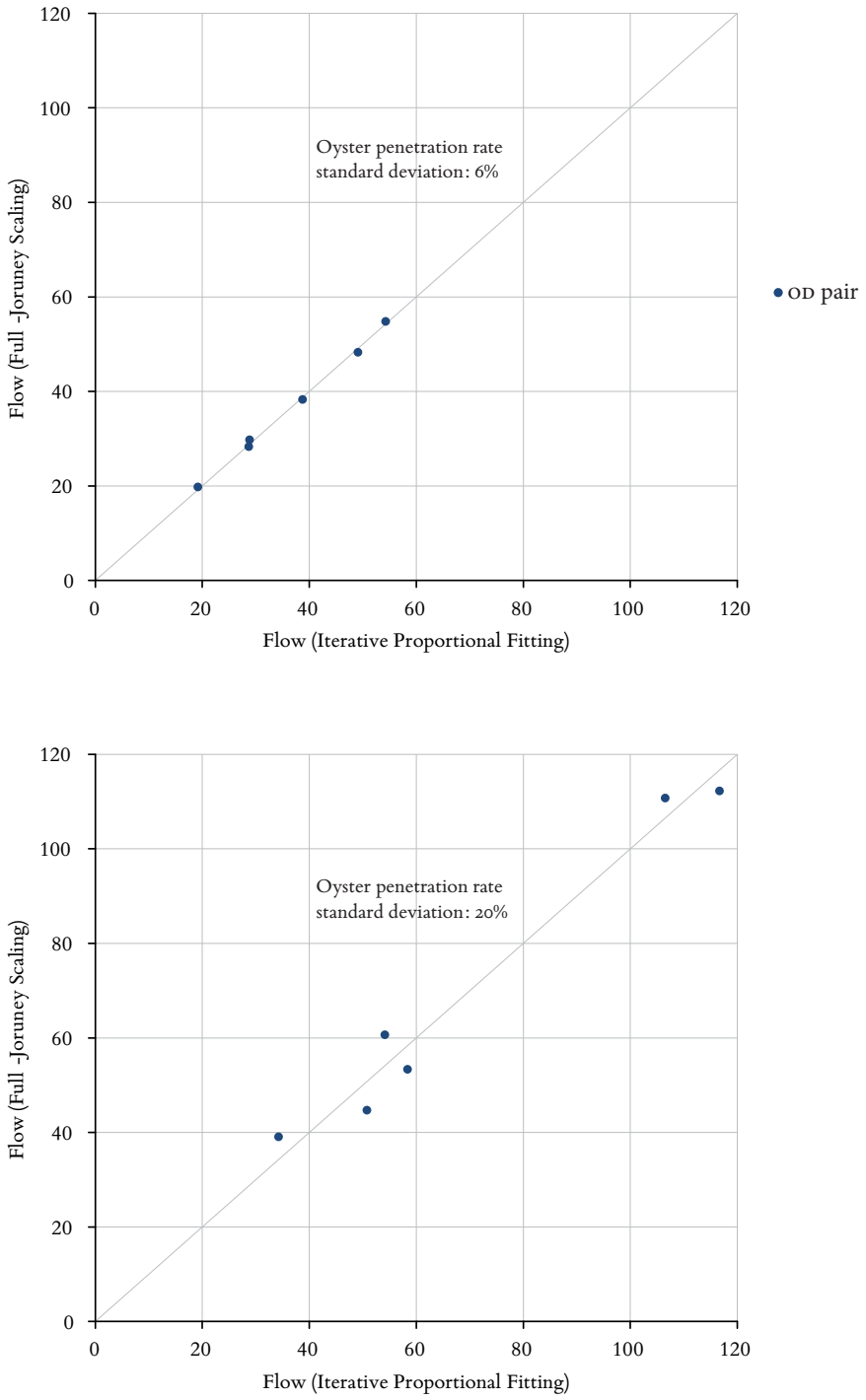


Figure 5-8. Comparison of passenger flows for the test network

Following its validation on the test network in section 5.5, the algorithm was implemented in a Java application (to be described in Chapter 6) and applied to the London network using five consecutive weekdays of Oyster and control-total data (October 17th–21st, 2011). Before calculating the scaling factors, all control totals are automatically adjusted as described in section 5.2.2. Figure 5-9 shows the results of this adjustment for Victoria Underground station. The distribution of station exits is shifted to the left, since some station exits correspond to journeys that started during an earlier hour. For many stations the entry counts are not shifted as far as the exit counts because station entries that are part of the first (or only) stage in a journey define the journey's start hour, thereby precluding any offset. Victoria Underground Station exhibits a relatively large shift in the distribution of entry counts because many of its customers transfer from its interconnected National Rail station in the mornings.[6]

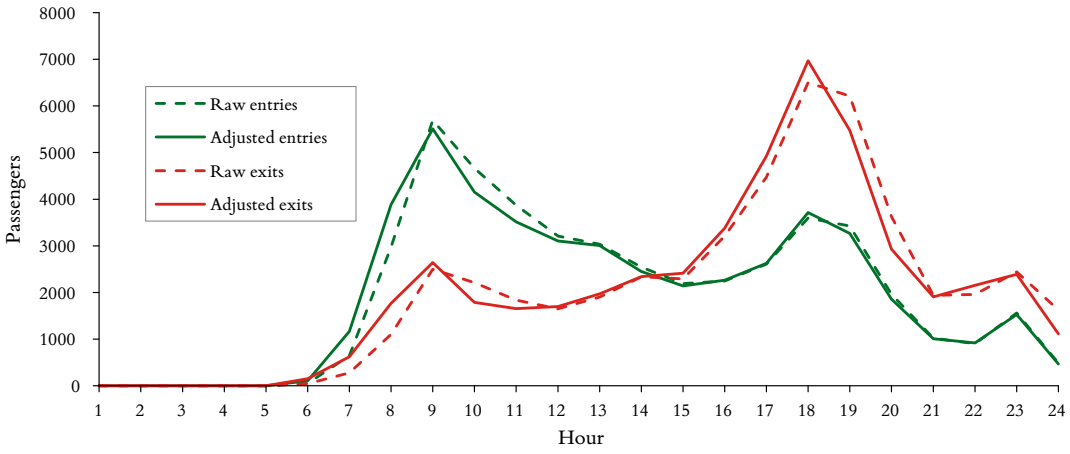


Figure 5-9. Hourly control totals for Victoria Underground Station, Wednesday, October 19, 2011.

6 The land uses surrounding Victoria Station are primarily commercial, yet the station's entries peak during the morning and its exits peak in the afternoon. While these peaks contain some number of nearby residents and interchanges from bus, the peaks are largely due to National Rail interchanges.

### 5.6.1 *Results*

After adjusting all entry and exit totals, the algorithm was applied to each of the five daily data sets until no itinerary scaling factors were adjusted by more than .001 percent (.00001) during a single iteration. For each of the five days, this condition was satisfied after approximately 450 iterations when generating a full-day OD matrix, and after approximately 2,500 iterations for peak-period matrices (with each iteration completing in less than one second). All bus and rail control totals were matched to within the ranges shown in Table 5-2.

Table 5-2. The degree to which the full-journey matrix expansion process satisfied the control totals for all Oyster transaction nodes (five-day total), as measured by the ratio of each node's scaled total to its control total.

| Transaction node type | Min. | Max. |
|---|---|---|
| Bus boarding totals (by route and direction) | 99.99999% | 100.00001% |
| Station entry totals | 99.99850% | 100.00468% |
| Station exit totals | 99.99537% | 100.00153% |

Unlike the test network, London's actual itinerary flows are unknown and cannot be compared with the estimated flows. Figure 5-10 shows the average daily distribution of full-journey scaling factors, as well as the ratios of bus and rail transaction node control totals to their sample totals. The distribution is shown for the AM peak period and for the entire day. Since each full-journey scaling factor (in the top histogram) is calculated as an average of the scaling ratios of its constituent transaction nodes (the middle and bottom histogram), it should be expected that the top distribution bears some similarity to the other two.

On average, the bus-route ratios are significantly higher than those for rail stations. This is partly due to riders who board buses without tapping an Oyster card, such as those who pay cash fares onboard or students whose passes are visually inspected by the operator (who can record the boarding to the ETM by pressing a button). The primary reason for this discrepancy, however, is that approximately 24 percent of Oyster bus transactions are excluded from the seed matrix because they were discarded during the origin- or destination-inference process. While rail

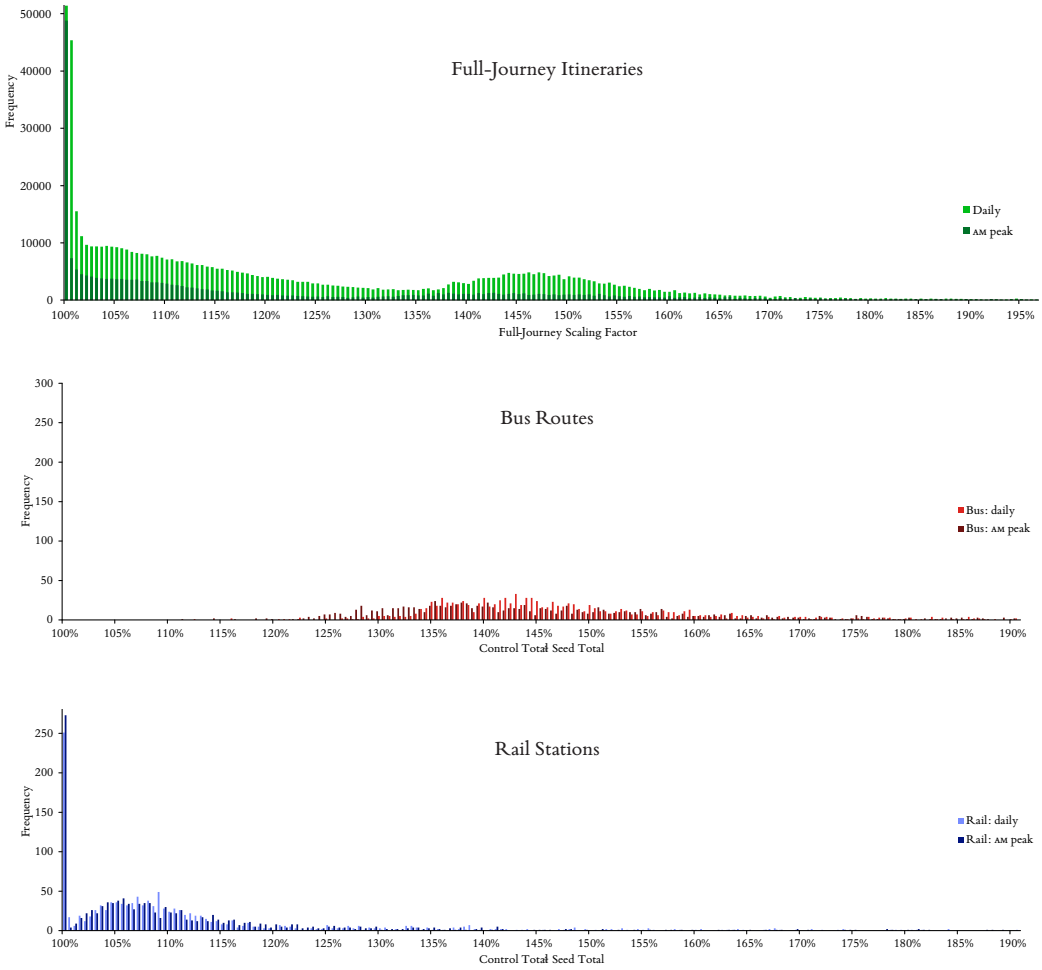


Figure 5-10. Histograms of scaling factors

scaling factors account for incomplete or non-Oyster activity, bus scaling factors account for non-Oyster activity plus the 24 percent of Oyster activity that was discarded. The histograms also show that bus ratios are lower during the AM peak period than throughout the entire day, suggesting that bus commuters are more likely to use Oyster cards than bus riders in general.

Although the distribution of gateline-to-Oyster ratios is significantly lower for rail stations, many of the stations with higher ratios have extremely high passenger flows. For example, four Underground stations—Victoria, Oxford Circus, Liverpool Street, and London Bridge—account for over ten percent of the total daily counts, and each station has a ratio higher than 1.4. Three of these stations are adjacent to National Rail facilities, and the fourth, Oxford Circus, is a popular destination from the other three. The higher ratios at these stations are likely due to commut-

ers who live outside Central London and use non-Oyster point-to-point season tickets to travel from National Rail to an Underground station near their workplace.

The large number of 100% scaling factors in the rail ratios is due to the fact that gateline data were not available for approximately 30 percent of rail station codes.[7] Most of these stations, however, belong to modes that exhibit relatively low ridership in the seed matrix or that have only recently been added to the Oyster network. Control totals should be obtained for these stations in the future, or their totals could be estimated using other methods before being input to the program.

At present, the stations lacking control totals have a relatively minor impact on the full-journey scaling process, as illustrated in Figure 5-11. The X and Y axes compare the sampled and scaled flows of each full-journey itinerary, and the itineraries that are scaled to approximately 100% (forming a 45-degree line from the chart's origin) can be seen to have relatively small flows.

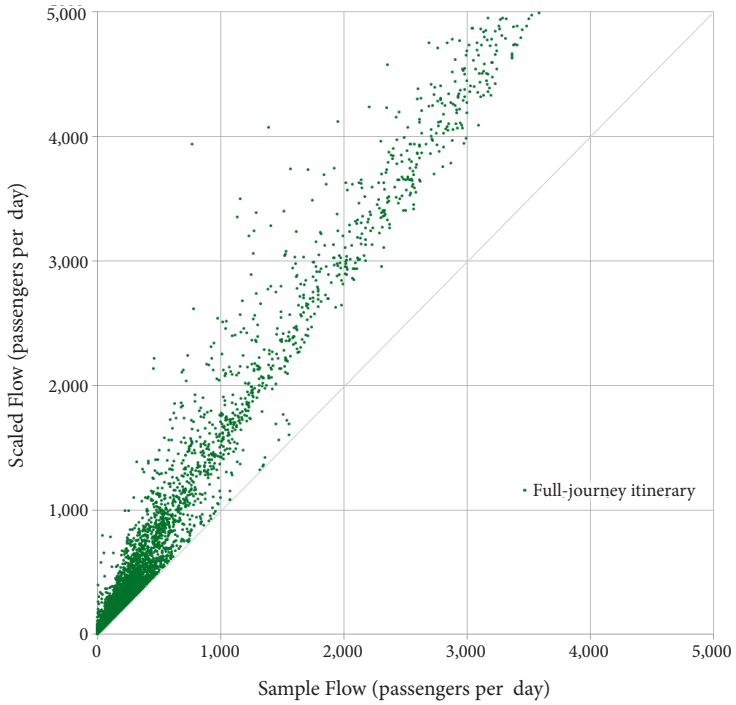


Figure 5-11. Comparison of sample (or seed) flows to scaled flows, for all itineraries (five-day average).

7 Gateline data were obtained for almost all Underground stations, but were not available for several National Rail stations and some newer Overground stations (although these data are recorded).

### 5.6.2 *Validation Against Iterative Proportional Fitting on the London Network*

As was done for the test network, a single-stage rail-only OD matrix was derived from the full-journey matrix for the London network, and this OD matrix was compared to another OD matrix generated using IPF with the same Oyster and control-total data. Full-day and AM-peak OD matrices were generated for each of the five weekdays, and the five-day average of each OD pair was used for comparison.

It was asserted in section 5.5.2 that the OD-pair scaling factors derived from the full-journey scaling process should vary more than those obtained from IPF, since the full-journey algorithm is subject to more constraints. The distribution of OD-pair scaling factors from both methods are shown in Figure 5-12: the series is sorted by scaling factor, with each point on the X axis corresponding to a single OD pair. The distributions are similar, but the full-journey curve exhibits greater variation, as it lies below the IPF curve throughout the left side of the graph and lies above it throughout the right. Since the curves in Figure 5-12 are sorted by scaling factor, however, the OD pair at any point on one curve is unlikely to be the same OD pair that occupies the same X position on the other curve.

Figure 5-13 compares both algorithms' scaled flows for each OD pair. While there is a generally linear relationship between the OD flows calculated using the two methods—presumably because both use roughly similar techniques to scale the same sample data to the same control totals—the variations between the two is expected, since the full-journey approach scales and constrains the problem at the itinerary level. The difference between the AM-peak and full-day scatter plots, however, is likely due to the smaller sample size of the shorter analysis period. While the

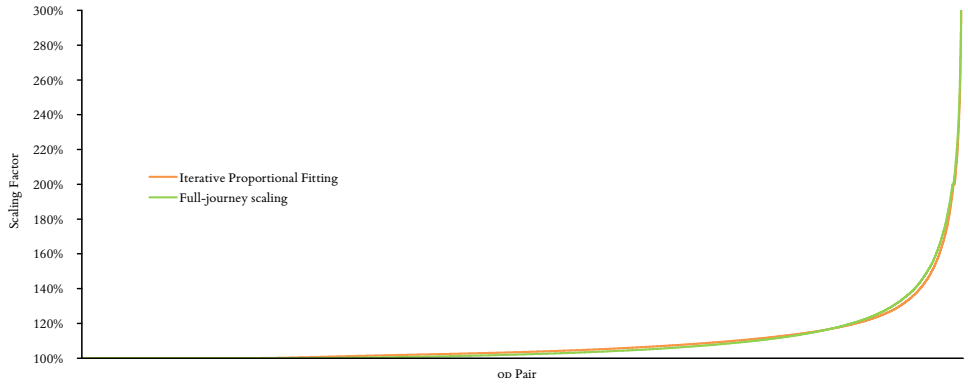


Figure 5-12. Distributions of OD-pair scaling factors on London's rail network.

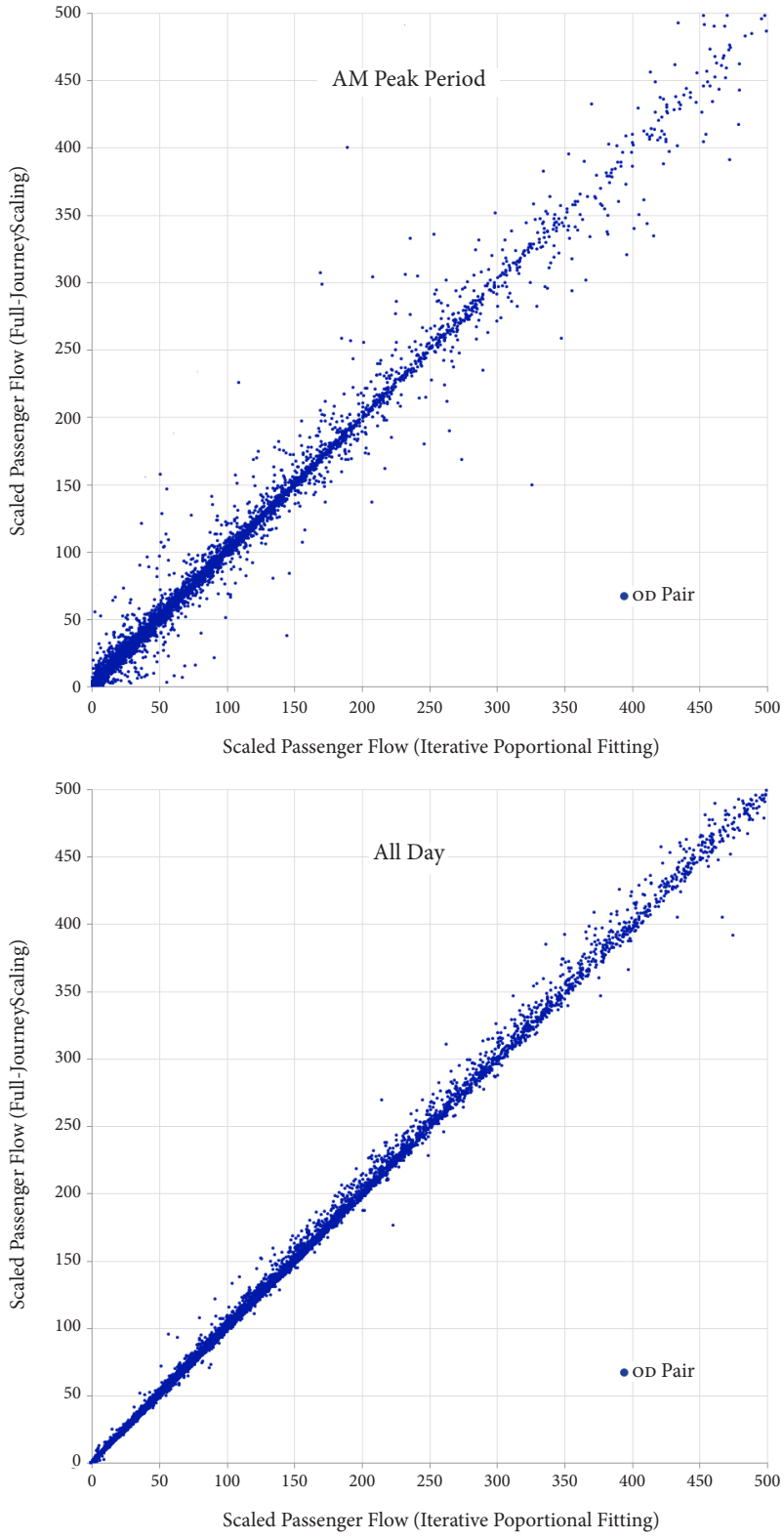


Figure 5-13. Comparison of full-journey scaling and IPF by rail OD pair.

program converged upon the control totals in both cases, the AM peak required roughly five times as many iterations to do so.

## 5.7 SUMMARY

Although the full-journey matrix-expansion process has not yet been tested against passenger surveys or other sources of empirical full-journey information, its performance on a small test network with known values, its matching of control totals on London's transport network, and the similarity between its constituent rail OD flows and those of a conventional OD matrix suggests that the algorithm is a reasonable method for estimating full-journey scaling factors. The algorithm might also be applicable to similar problems, such as the estimation of highway flows by using survey data to construct a seed matrix and traffic counts to serve as control totals.

While this method provides a novel solution to the problem of scaling full journey itineraries, transport agencies still perform many analyses at the level of the unlinked passenger trip. By applying the full-journey expansion process to a transport network and storing the scaling factors or the scaled route- or OD-level flows in a central database, single- or multi-stage analyses can be performed throughout the agency in a way that ensures compatibility between their outputs.

The technical implementation of the full-journey expansion algorithm is discussed in Chapter 6, and multiple applications of the process are presented in Chapter 7.

# Implementation

# 6

One of the primary advantages of using automatically collected data as a source of public-transport passenger information is the unprecedented size of the sample. AFC, AVL, and aggregate transaction-count data enable transport providers to observe the travel activity of most of their customers on any given day, but this data is only useful if it can be processed efficiently. One of the goals of this thesis is to demonstrate the feasibility of processing complete sets of London's Oyster and iBus data on a daily basis, and a significant portion of this research was devoted to the design of a software application that achieves this goal.

This chapter describes the implementation of the origin-, destination, and interchange-inference processes, as well as that of the full-journey matrix-expansion algorithm. While exhaustive documentation of the software's code is beyond the scope of this thesis, the goal of this chapter—supplemented by the methodologies presented in chapters 3 through 5—is to provide enough information to guide the development of similar implementations.[1]

## 6.1 OVERVIEW

The algorithms developed in this thesis were implemented using the Java programming language, chosen for its flexibility, cross-platform compat-

1 Users of the code developed for this thesis, or those seeking additional technical documentation, should contact the author or the MIT Transit Research Group.

ibility, and succinctness. Rather than using a database to execute these procedures, the program takes advantage of Java's object-oriented features and stores collections of data objects in memory (and occasionally caching them to disk), providing more control over the process and the allocation of the system's resources.

The application, consisting of approximately 10,000 lines of code, was developed and tested on a consumer-grade PC to demonstrate that a specialized server is not needed to execute the application,[2] and care was taken to minimize the memory footprint and tune the performance of the process. A high-level overview of the process and its inputs and outputs is shown in Figure 6-1.

The classes of the software package are shown in Figure 6-2. The origin-, destination-, and interchange-inference processes are performed by the `OysterTransactionProcessor` and `NearestStopCalculator` classes, while full-journey matrix expansion is performed by the `ODXMatrixGenerator` class. Journey-stage data are represented by the

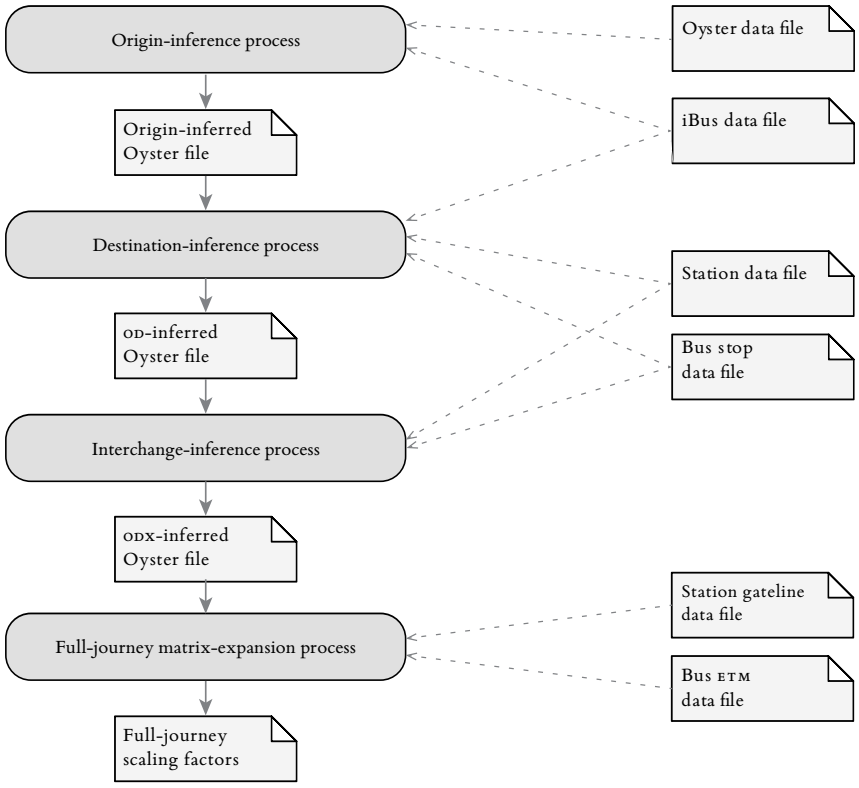


Figure 6-1. Overview of processes, inputs, and outputs.

2 Unless otherwise noted, tests were performed on a PC with a 2.8 GHz Intel i7 Core processor and 8 GB of RAM, running the Windows 7 operating system.

subclasses of both `OysterTransaction` and `Stage`: the former represents a relatively complete copy of the Oyster input data while the latter contains a lightweight version having only those fields that are relevant to the linking of journey stages. The `OysterTransactionProcesser` dispatches searches for iBus data to an `IBusManager`, which contains a hierarchical data structure of routes, vehicle trips, and stop events, while a collection of `OysterCard` objects are used to store passengers' daily travel histories and perform various sorts, searches, and enumerations on their constituent stages.

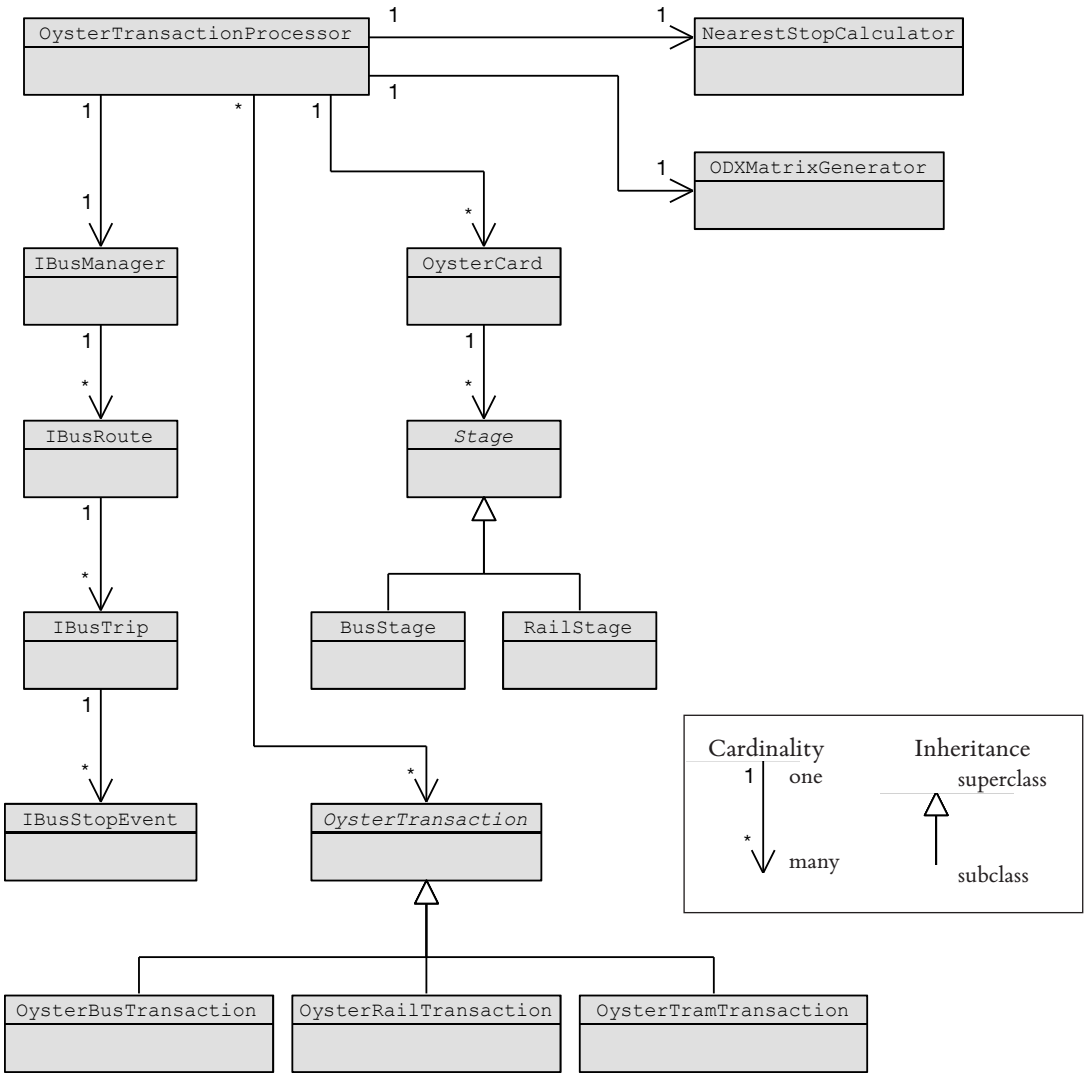


Figure 6-2. Class diagram of origin-, destination-, and interchange-inference package

To maximize compatibility with the program's various data sources and the systems which ultimately process its outputs, the application loads data from text files and generates output in a similar format (although the program could also be configured to interface directly with databases). The program can be divided into five general processing steps, which are described in the following subsections.

### 6.2.1 *Step 1: Preprocessing and Nearest-Stop Calculation*

The largest input to the program is the Oyster data set, which typically consists of roughly 16 million daily records, but which contains several fields that are not used by the program. Rather than storing these data in memory, only the relevant fields are retained and the full records (including the information inferred by the program) are temporarily cached to disk. All other data which must be matched with the Oyster records—such as spatial information and AVL data—are therefore loaded into memory prior to the processing of Oyster information so that they can be looked up as each record is read.

The first step in this process, as shown in Figure 6-3, is the loading of all bus-stop IDs and their spatial coordinates. iBus data, typically comprising approximately 5 million records per weekday, are then loaded into memory from a text file and are stored in the `IBusManager` data structure, with each record from the input file being stored in an `IBusStopEvent` and added to the appropriate parent `IBusTrip` and `IBusRoute`. Each vehicle-trip number is unique within a route and day, thus allowing Oyster records to be uniquely associated with a trip using three fields (day, route, and trip).

While the iBus data are being loaded and organized, an additional data structure is built containing spatial information for each route pattern. The top level of this second data structure consists of route patterns, each uniquely identified by its route name and pattern number, separated by a period (for example, "24.4882" denotes route 24, pattern 4882). The second level, within each pattern, contains the codes of all bus stops in that pattern.

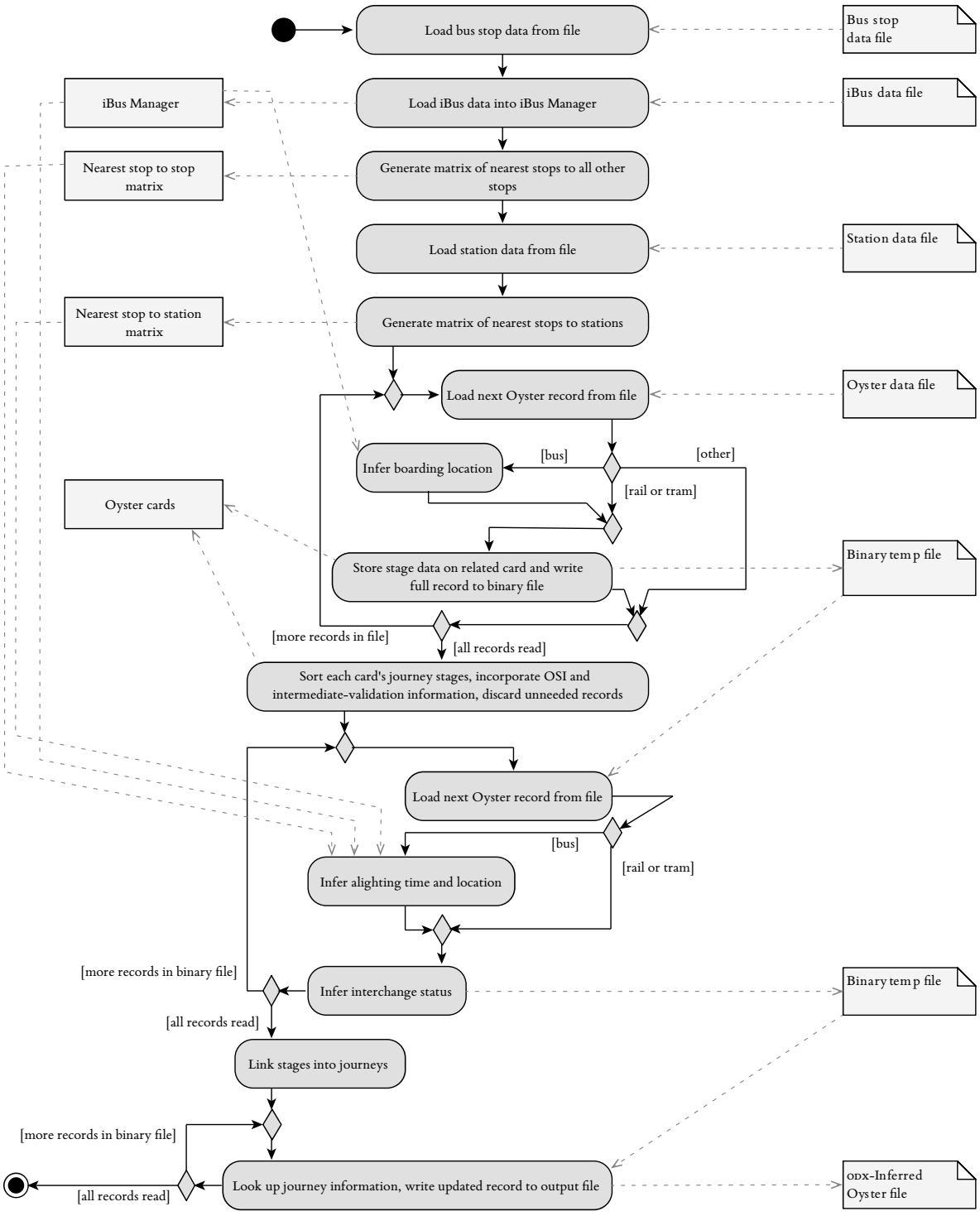


Figure 6-3. Origin-, destination-, and interchange-inference implementation.

The pattern collection is loaded in parallel with the stop-event collection: each stop event is denoted by its route and pattern, and redundant stop events are ignored (since many bus vehicle trips serve the same pattern). Finally, spatial coordinates are loaded for all bus stops and rail stations, enabling the calculation of distances during this and other steps of the destination- and interchange-inference algorithms.

The destination-inference process requires knowledge of the closest stop on a bus pattern to a cardholder's subsequent bus boarding or station entry location. Rather than calculating the nearest stop for each of the approximately six million daily bus alightings (which requires a distance calculation for each stop in a route's pattern), nearest stops are calculated once for each combination of route and subsequent location, and are stored for later reference. (The resultant "nearest-stop" matrices—one for subsequent bus stops and another for subsequent rail stations—are cached to disk and can be reused for multiple days' analyses for as long as the network and route patterns remain unchanged).[3] Station information is therefore loaded prior to the generation of the nearest-stop-to-station matrix, and is retained in memory for use by other steps of the destination- and interchange-inference processes.

Efficient processing and memory allocation are top priorities due to the large sets of data required. All four matrices are therefore implemented as two-dimensional primitive arrays, and the indices of the rows and columns are implemented as one-dimensional arrays sorted by their values to enable efficient access using the binary search algorithm (Knuth 1973).

### 6.2.2 *Step 2: Origin Inference and Daily History Reconstruction*

After iBus and spatial data are loaded into memory and initialized, the program makes its first of three iterations over the Oyster data set to infer bus boarding locations while storing compact versions of the Oyster records (stored as a subclass of `Stage`) to the appropriate `OysterCard` object. For memory efficiency, the Oyster file is read one record at a time, looking up each bus transaction's boarding location in the iBus data structure and writing it to a binary output file (from where it will be read and

3 Bus stop and pattern information are typically updated every two weeks in iBus, or sooner if changes are implemented.

processed during the following iteration).[4] Tram boardings and station exits are written to the output file without being processed, since their origins are already known.

Oyster record types that are not relevant to the program—such as top-ups, voids, unfinished station entries, and unstarted station exits—are ignored, along with duplicate transactions made with TfL staff cards. In many cases TfL staff use their Oyster cards to open station gates for exiting customers, often in response to equipment malfunctions, to correct erroneous transactions, or to expedite passenger egress. All of these exit transactions are assigned the entry time and location of the staff member's previous entry tap and, moreover, the transactions are recorded to the staff member's card rather than the rider's. For these reasons, any exit transaction made on a staff card but not preceded by an entry transaction is discarded (these discarded records should be accounted for in the control totals and therefore the scaling process).

Special handling is required for out-of-station interchanges (OSIS, described below) and intermediate-validation transactions. Intermediate-validation and completed station-entry records are therefore temporarily stored, along with the record types copied to the binary output file, as `Stage` objects in the associated `OysterCard`'s collection.

After all records in the Oyster input file have been processed, each `OysterCard` object sorts its child `stage` objects by transaction time to construct a daily travel history. Intermediate-validation and OSI information are then added to the appropriate journey stages before performing the second iteration over the Oyster data

OSIS are designated pairs of stations at which cardholders can tap out of one station, quickly tap into the other, and pay a single fare for the combined journey. When OSIS are recorded, the start time and location of the stage before the interchange is written to its own exit record as well as that of the stage after the interchange (recall from section 2.2.1 that entry records contain entry data only, while exit records contain both entry and exit data in order to charge the correct fare). This is appropriate for the fare-collection application for which the Oyster system was designed: no fare is charged for the prior stage, while the latter stage denotes the origin, destination, and full fare of the combined journey. For the purpose of

4 The records are read from and written to memory buffers, to minimize the number of harware I/O transactions.

full-journey travel analysis, however, it is important to accurately represent each physical stage in the cardholder's journey. Therefore, the station entry and exit records immediately following the OSI are compared, and the start time and location are copied from the start record to the exit record so that both journey stages associated with the OSI contain their physical entry locations and times. The temporary `Stage` objects representing station entries are then discarded.

Following the handling of OSIs, the program incorporates intermediate-validation information, recorded by a dedicated type of Oyster record which indicates the time and location of a cardholder transfer "behind the gate."[5] Since no other type of behind-the-gate transfers are recorded by the Oyster system, intermediate validations are not retained by the software as fare transactions. However, they provide useful information about the cardholder's path and are therefore appended to two optional fields in the related Oyster journey-stage record.

### 6.2.3 *Step 3: Destination and Interchange Inference*

Only after bus origins have been inferred and travel histories constructed can interchanges or bus destinations be inferred. To ensure that all fields in the Oyster input file are present in the output, the temporary binary file written during the previous step is processed sequentially, and interchanges and bus alightings are inferred by looking up the associated travel history in the collection of `OysterCard` objects. As in the first iteration, the updated Oyster records are written to a buffered, temporary binary file.

Bus alighting times and locations are inferred using the algorithm discussed in section 3.3.2. As each Oyster bus record is read, its corresponding journey stage object is retrieved from memory along with that of the following stage (specified by the `OysterCard` object), and the specified route, pattern, and target location data are used to retrieve the nearest stop and its distance by looking up its value in the nearest-node matrices. The inferred destination data are additionally recorded to the associated `BusStage` object, in preparation for the inference of interchanges.

---

5 Intermediate validators enable customers to tap their cards at designated interchange stations to show that they transferred in a travel zone with a lower fare than that of another potential interchange location.

After each bus destination is inferred, or rail record is read, its interchange status is inferred (tram journey stages can be linked to, and they can be used to infer preceding bus destinations, but their interchange status cannot be inferred because their own destinations are unknown[6]).In both the binary output file and the `OysterCard` data structure, alighting times and locations are recorded, as well as the times and distances to the cardholder's next transaction and a flag indicating whether the stage was linked to the following one. If a bus stage's destination or any stage's interchange status cannot be inferred, these values are replaced with one of several error codes to indicate which test failed.

### 6.2.4 *Step 4: Full-Journey Construction*

The preceding step inferred whether each stage was linked to the next, but stages cannot be linked into full journeys until the interchange status of all of a card's stages have been inferred.[7] For this reason, journeys are constructed after the second Oyster iteration, and a third and final iteration is required to append this journey information to the full Oyster records while writing them to the output file.

After the second Oyster iteration, all `OysterCard` objects are instructed to link their child `Stage` objects into journeys. By inspecting the interchange-status flag of each stage, each can be assigned a journey number, a stage number within the journey, and the number of stages in the journey (for example, journey 2, stage 2 of 3). These three fields enable the output data to be generated in a format similar to that of the input data (with each row representing a journey stage rather than a full journey), which in turn enables the full-journey matrix expansion software (or other analysis tools) to easily join records into journeys while retaining the details of each stage.

After recording journey information to each `Stage` object, the second binary file is read sequentially, the journey information is retrieved from memory, and data from both are combined and written to the final Oys-

---

6 Tram transactions note the station where the fare was paid but riders do not tap their cards upon alighting. Tram destinations could be inferred using vehicle-location data, but this was not available.

7 Oyster data are not guaranteed to be stored in temporal order in the original input file or the subsequent binary files.

ter output file. The file is identical in format to the original Oyster input file, but contains the additional fields generated during the inference processes. The file can then be loaded into a database or other analysis tool, or can be used to construct intermodal full-journey matrices.

### 6.2.5 *Step 5: Full-journey Matrix Expansion*

The full-journey matrix expansion process was implemented in the `ODXMatrixGenerator` class, which can be called independently from the `OysterTransactionProcessor`, provided that the latter has been run and has generated the necessary Oyster output file. Following the methodology presented in Chapter 5, the application generates an intermodal full-journey matrix for one or more time periods specified by the user, adjusts the network's control totals, and estimates expansion factors to account for unobserved or uninferred passenger flows.

The first step is the construction of the seed matrix, as illustrated in Figure 6-4. The ODX (origin, destination, and interchange)-inferred Oyster file is parsed and the origin and destination stop or station codes of each record are stored in a lightweight inner class representing a journey stage, with each stage being loaded to a similarly lightweight object representing a full journey. After Oyster data are loaded, all journey objects are inspected and a set of full-journey itinerary identifiers is constructed by concatenating the identifiers of the transaction nodes that comprise the journeys. Since expansion factors are calculated at the bus-route level, but full-journey matrices are to be constructed at the stop level, two sets of matrices and identifiers are generated. For example, the identifier `_24.708/_24.796/755/540` would signify all customers who boarded bus route 24 at stop 708, alighted at stop 796, transferred to station 755, and ended their journeys at station 540.[8] The identifier for the associated route-level itinerary would be `_24/755/540`.

The route- and stop-level seed matrices are implemented as four-dimensional integer arrays indexed by transaction node, recording period, span, and direction (inbound vs. outbound, or entry vs. exit).[9] The first dimension of each array is then mapped to the itinerary identifiers, and all

8 Underscores denote bus transaction nodes, periods separate routes from stops, slashes separate transaction nodes, and tildes denote tram stations.

9 See section 5.2.2 for terminology

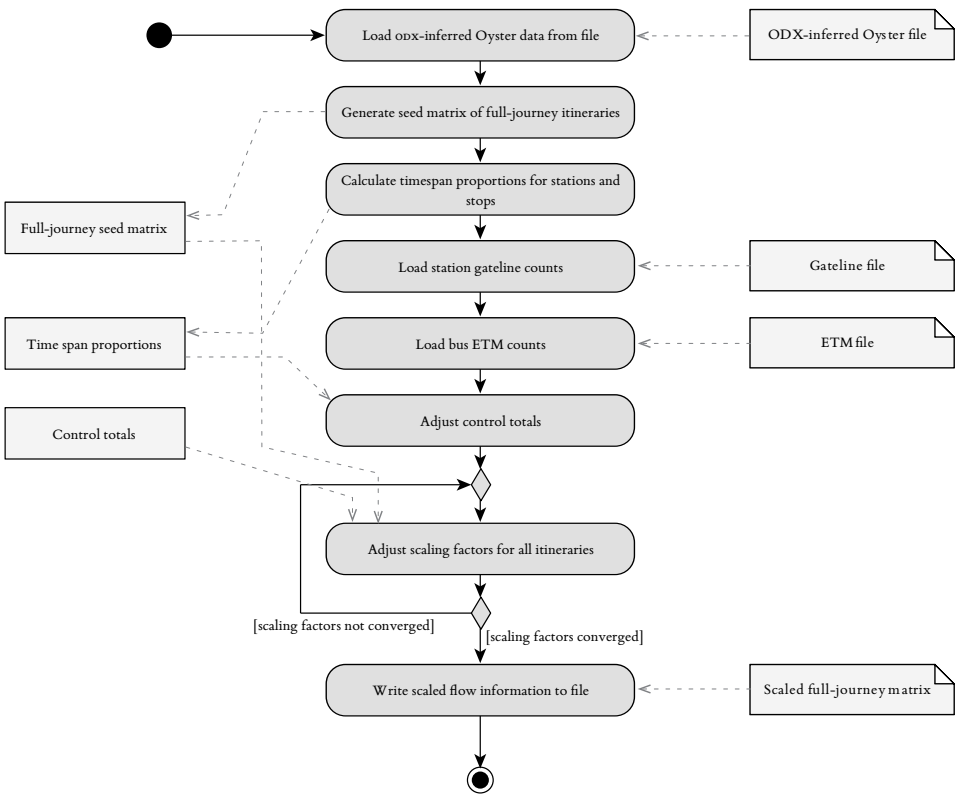


Figure 6-4. Full-journey matrix-expansion implementation.

journeys are inspected again to look up the appropriate elements in both seed matrices and increment the appropriate count. The journey collection is then discarded and the memory allocated to it is cleared.

Once the seed matrices are constructed, timespan proportions are initialized by constructing a four-dimensional array of floating-point variables with the same indices as the route-level seed matrix. The route-level seed matrix is then analyzed as described in Chapter 5 to calculate the timespan-proportion values.

Gateline and ETM data are then loaded from text files, in which they are aggregated by node, hour, and direction (inbound/outbound for bus, entry/exit for rail).[10] Each input record is multiplied by the appropriate timespan proportion and is added to the corresponding element of a three-dimensional array, similarly indexed by node, hour, and direction.

10 TfL's gateline and ETM data are provided as hourly counts, which relate to the recording periods in this implementation.

Finally, the scaling factors are estimated by repeatedly performing the calculations in equations 5.9 and 5.10. The application iterates over the entire adjustment process until the greatest change in any node's scaling factor between two successive iterations is less than the user-defined *convergence threshold* parameter, or until the number of iterations exceeds the *maximum iterations* parameter.

## 6.3 RESULTS AND PERFORMANCE

The origin-, destination-, and interchange-inference processes, which are executed in a single function call, typically complete within 20 minutes on an eight-core, 2.8 gigahertz, eight-megabyte PC. TfL staff have run the process in under eight minutes on a 32-core 2.9 gigahertz server with 256 gigabytes of RAM. The full-journey matrix-expansion process, which includes the construction of seed matrices and the adjustment of control totals, typically completes in less than ten minutes on the MIT machine and three minutes on a TfL server.

In addition to the text files containing the processed Oyster and full-journey matrix information, the application generates a series of reports that contain the performance statistics and inference rates displayed in tables 3-1, 3-2, 3-4, 4-3, and 5-2.

## 6.4 SUMMARY

The performance of the software described in this chapter demonstrates the feasibility of applying the methodologies of this research on a daily basis. By automating the application to run after hours, when all daily AFC, AVL and transaction-count data have been transferred from their respective servers, transport providers could obtain continuous system-wide full-journey information. These data could be used to generate a variety of automated reports, and would enable ex post analyses of any portion of the network at any time.

# Applications

7

The methods developed in this thesis can be applied to a range of analyses on multiple public transport modes at various spatial and temporal scales. TfL staff have used Oyster data to perform many analyses on the Underground, Overground, and National Rail networks (due to the requirement that passengers tap their cards both upon entering and exiting the system), but the inference of origins and destinations on London's buses allows TfL's highest-ridership mode to be included in these analyses.[1] In addition to observing travel on these multiple modes, the inference of interchanges enables the observation of passengers' full journeys, both within and between modes.

This chapter demonstrates some of the applications of the methods developed in this thesis, and describes its use by other researchers and TfL staff. Analyses that rely only on the outputs of the OD-inference process are presented first, followed by a summary of full-journey and longitudinal applications.

1 Bus ridership is higher than other public modes in terms of the number of Oyster journey stages per day

### 7.1.1 *Boarding/Alighting/Flow Profiles and Route-Level OD Matrices*

By extending the Oyster data set to include bus boardings and alightings, the origin- and destination-inference process enables a number of analyses independently of the interchange-inference and full-journey expansion methods. Since passengers must tap their Oyster cards when boarding each bus, analyses can be conducted at the vehicle-trip level or can be aggregated over many trips at the route level.

Enriched Oyster bus data can be used to generate passenger boarding, alighting, and flow profiles, as demonstrated in Figure 7-1 (showing route 488 inbound, 7–10 AM). After loading the updated Oyster data into a database, records can be aggregated by boarding and alighting locations, with the cumulative difference between the two counts (boardings minus alightings) revealing the passenger flow. The figure illustrates the total flow on the route for five consecutive weekdays (9–13 May 2011) during the AM peak, but individual vehicle loads can be calculated by simply aggregating the data by vehicle trip. Doing so could be useful for observing the effects of crowding on rider behavior, dwell time, or service reliability.

In addition to aggregating ridership information by boardings and alightings as in Figure 7-1, the data can be more finely aggregated by OD pair. Figure 7-2 shows an OD matrix for the same route, direction, and time period as shown in the graph, enabling a better understanding of the relationships between the route's boarding and alighting counts.

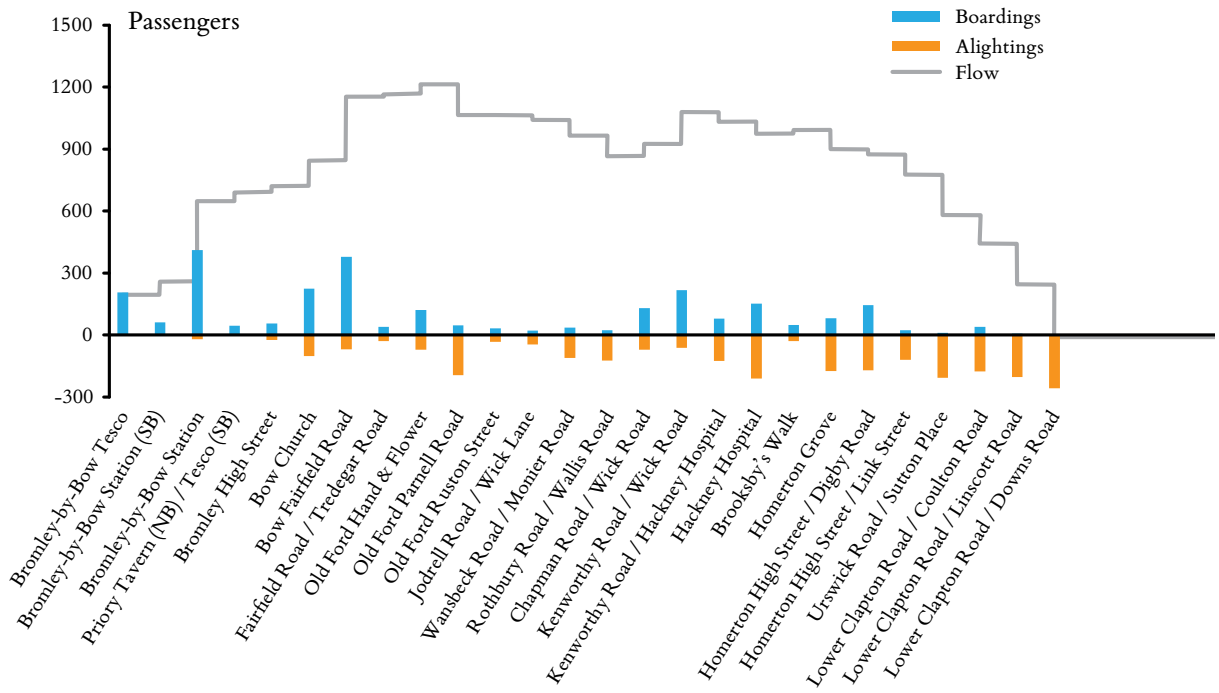


Figure 7-1. Boarding/alighting/flow profile for Route 488 inbound, 7–10 AM

DESTINATION

| ORIGIN | Bromley-by-Bow Tesco | Bromley-by-Bow Station (SB) | Bromley-by-Bow Station | Priory Tav. (NB) / Tesco (SB) | Bromley High Street | Bow Church | Bow Fairfield Road | Fairfield Road / Tredegar Road | Old Ford Hand & Flower | Old Ford Parnell Road | Old Ford Ruston Street | Jodrell Road / Wick Lane | Wansbeck Road / Monier Road | Rothbury Road / Wallis Road | Chapman Road / Wick Road | Kenworthy Road / Wick Road | Kenworthy Road / Hackney Hospital | Hackney Hospital | Brooksby's Walk | Homerton Grove | Homerton High Street / Digby Road | Homerton High Street / Link Street | Urswick Road / Sutton Place | Lower Clapton Road / Coulton Road | Lower Clapton Road / Linscott Road | Lower Clapton Road / Downs Road | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bromley-by-Bow Tesco | – | 0 | 17 | 0 | 7 | 24 | 11 | 4 | 7 | 15 | 7 | 7 | 2 | 2 | 9 | 4 | 0 | 4 | 2 | 2 | 0 | 0 | 2 | 4 | 0 | 0 | 130 |
| Bromley-by-Bow Station (SB) | – | – | 0 | 0 | 2 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 2 | 7 | 0 | 0 | 0 | 0 | 0 | 15 |
| Bromley-by-Bow Station | – | – | – | 2 | 9 | 30 | 28 | 2 | 15 | 35 | 7 | 9 | 56 | 9 | 11 | 11 | 37 | 13 | 4 | 46 | 35 | 0 | 7 | 7 | 9 | 4 | 384 |
| Priory Tav. (NB) / Tesco (SB) | – | – | – | – | 2 | 20 | 2 | 0 | 0 | 4 | 2 | 2 | 2 | 0 | 2 | 0 | 2 | 4 | 0 | 2 | 4 | 0 | 0 | 2 | 0 | 4 | 56 |
| Bromley High Street | – | – | – | – | – | 4 | 4 | 0 | 0 | 11 | 2 | 7 | 0 | 2 | 0 | 0 | 2 | 2 | 0 | 9 | 0 | 2 | 0 | 0 | 0 | 0 | 46 |
| Bow Church | – | – | – | – | – | – | 7 | 7 | 17 | 30 | 4 | 0 | 7 | 7 | 2 | 4 | 4 | 17 | 2 | 2 | 9 | 7 | 9 | 4 | 2 | 4 | 146 |
| Bow Fairfield Road | – | – | – | – | – | – | – | 7 | 13 | 48 | 9 | 7 | 15 | 15 | 9 | 13 | 28 | 26 | 2 | 9 | 11 | 2 | 0 | 0 | 2 | 9 | 224 |
| Fairfield Road / Tredegar Road | – | – | – | – | – | – | – | – | 4 | 0 | 0 | 0 | 0 | 11 | 0 | 0 | 0 | 2 | 0 | 4 | 4 | 0 | 2 | 2 | 0 | 0 | 30 |
| Old Ford Hand & Flower | – | – | – | – | – | – | – | – | – | 0 | 0 | 2 | 0 | 9 | 7 | 2 | 2 | 20 | 0 | 11 | 13 | 9 | 0 | 2 | 2 | 11 | 89 |
| Old Ford Parnell Road | – | – | – | – | – | – | – | – | – | – | 0 | 4 | 0 | 2 | 7 | 4 | 0 | 4 | 2 | 4 | 2 | 0 | 2 | 0 | 0 | 4 | 37 |
| Old Ford Ruston Street | – | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 17 | 2 | 0 | 0 | 4 | 0 | 2 | 0 | 0 | 4 | 2 | 0 | 7 | 39 |
| Jodrell Road / Wick Lane | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 11 | 2 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 2 | 0 | 0 | 20 |
| Wansbeck Road / Monier Road | – | – | – | – | – | – | – | – | – | – | – | – | – | 7 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 0 | 2 | 0 | 4 | 17 |
| Rothbury Road / Wallis Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 2 | 0 | 0 | 2 | 7 |
| Chapman Road / Wick Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 7 | 9 | 9 | 2 | 13 | 15 | 9 | 11 | 15 | 24 | 22 | 135 |
| Kenworthy Road / Wick Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 9 | 43 | 7 | 13 | 15 | 13 | 35 | 17 | 28 | 26 | 206 |
| Kenworthy Road / Hackney Hospital | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 7 | 0 | 2 | 0 | 4 | 4 | 4 | 17 | 13 | 52 |
| Hackney Hospital | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 4 | 15 | 13 | 15 | 35 | 20 | 22 | 124 |
| Brooksby's Walk | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 2 | 0 | 4 | 15 | 7 | 11 | 4 | 43 |
| Homerton Grove | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 7 | 13 | 15 | 7 | 15 | 61 |
| Homerton High Street / Digby Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 17 | 35 | 24 | 13 | 24 | 113 |
| Homerton High Street / Link Street | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 2 | 0 | 4 | 9 | 15 |
| Urswick Road / Sutton Place | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 7 | 7 |
| Lower Clapton Road / Coulton Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 9 | 2 | 11 |
| Lower Clapton Road / Linscott Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 2 | 2 |
| Lower Clapton Road / Downs Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 |
| | 0 | 0 | 17 | 2 | 20 | 78 | 52 | 22 | 56 | 143 | 30 | 37 | 83 | 91 | 52 | 46 | 93 | 156 | 24 | 128 | 139 | 91 | 159 | 146 | 148 | 195 | 2009 |

Figure 7-2. Origin–destination matrix for Route 488 inbound, 7–10 AM

### 7.1.2 *Bus Passenger Travel Time and Distance*

In London, revenue is allocated to bus operating companies according to the number of passenger miles traveled on each route, calculated as the number of passengers recorded by the ticket machine multiplied by that route's average passenger journey-stage length. TfL has traditionally used the Greater London Bus Passenger Survey (GLBPS) to determine average journey length, but analysts in the organization's Fares and Ticketing group have been testing the software developed in this thesis as a possible replacement.[2]

Figure 7-3 plots the average journey distance observed through GLBPS against those derived from the output of the OD-inference process. Each

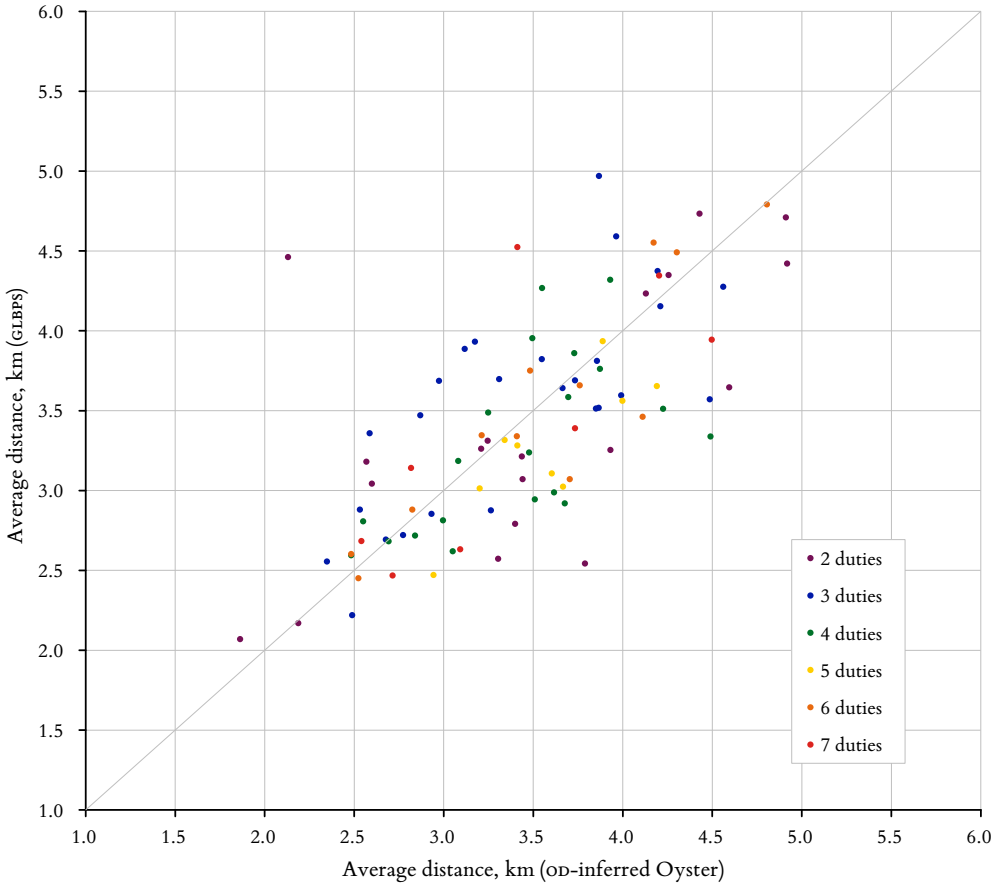


Figure 7-3. Comparison of average journey-stage lengths: GLBPS vs. OD-inferred Oyster

2 Bus passenger miles were similarly inferred from AFC data by Lu and Reddy (2011).

point corresponds to a single bus route, and the data are classified by the number of GLBPS duties—or surveyor shifts—that were undertaken on each route. Routes with more GLBPS duties matched the Oyster data more closely: Figure 7-4 shows the distributions of error between Oyster and GLBPS, classified by number of duties (the lines extending from each class indicate the minimum and maximum error, while the bottom, middle, and top lines of the boxes indicate the first, second, and third quartile error values). For routes having zero or one GLBPS duties, TfL estimates an average journey length by assigning the average of other routes having the same fare zone (hence, an average of average journey lengths). Larger sample sizes (in terms of numbers of GLBPS duties) appear to match the Oyster average journey lengths more closely, as the median error for five-duty routes is ten percent while 75 percent of routes having six or more duties have less than a ten percent error.

Similarly to the calculation of bus journey-stage distances, durations can be easily distilled from OD-inferred Oyster bus data. Schil (2012) uses the outputs of the OD-inference process to calculate bus passengers' in-vehicle travel times, which he then analyzes as an indicator of service reliability and expected travel time. By calculating the same metrics for rail

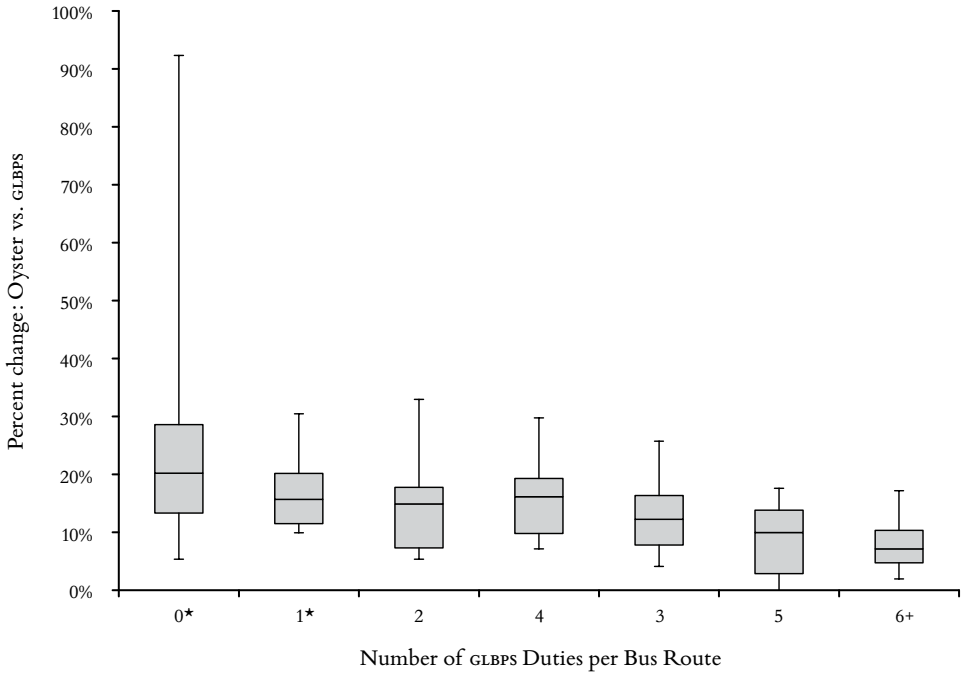


Figure 7-4. Comparison of average journey-stage lengths by bus route: GLBPS vs. OD-inferred Oyster

stages, he is able to propose a single set of reliability metrics across all TfL modes and for full intermodal journeys.

## 7.2 CROSS-MODAL AND FULL-JOURNEY APPLICATIONS

The applications discussed in the previous section were centered on buses, although similar analyses have also been applied to London's rail modes. By inferring interchanges, however, travel can be studied across modes, including the activity of passengers who span multiple modes in a single journey.

The 7 million daily full passenger journeys revealed by the interchange-inference process typically follow approximately 1.7 million different full-journey itineraries. Full journey matrices are therefore cumbersome to present in their entirety, but can be aggregated to reveal the flow to or from specific nodes or zones. For example, the origins of all full journeys destined for Oxford Circus Underground station during an AM peak period are shown in Figure 7-5.[3] The busiest AM destination on the Oyster rail network, Oxford Circus (highlighted in yellow, near the city's center) can be seen to attract riders from throughout Greater London. Origin node counts are color coded to denote the mode of the first stage: all bus origins (red) therefore entail transfers to rail lines, since all activity on the map terminated at an Underground station.

Rather than indicating origins and destinations at the level of individual bus stops or rail stations, these nodes can also be spatially aggregated by zones—such as postcodes, census areas, or traffic analysis zones. Additionally, intermodal full-journey matrices can be aggregated by time, enabling the visualization of the entire network's travel activity.

Figure 7-6 captures a frame of an animated time-lapse full-journey matrix (Gordon 2011). An optional function built into the matrix-expansion module tracks all passenger flows over the course of a day and interpolates each cardholder's location every minute. Customers' activities are broadly inferred, as each passenger is either in a transit vehicle, between transit trips (and therefore either performing an activity or transferring), or at home (before their first or after their last journey of the day). Such

3 7–10 AM, Wednesday, 19 October 2011

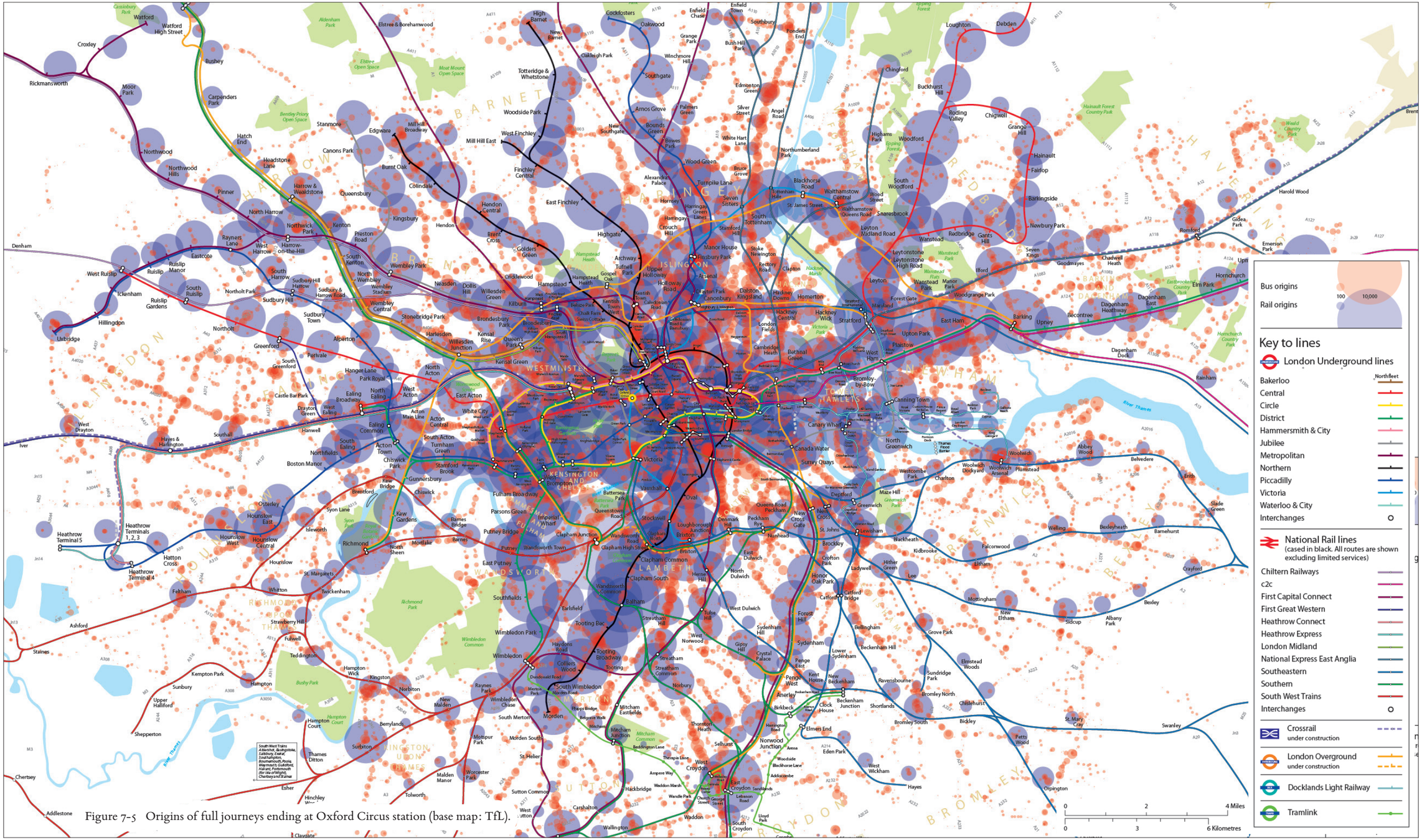


Figure 7-5 Origins of full journeys ending at Oxford Circus station (base map: TfL).

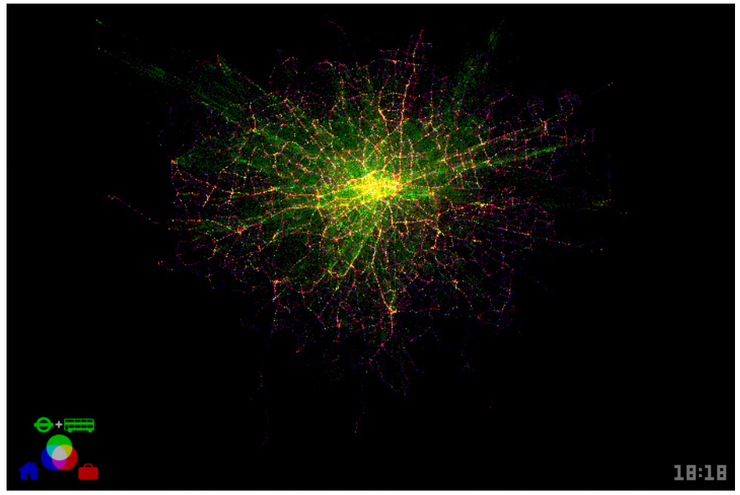


Figure 7-6. Frame from a time-lapse animation of a full day's inferred Oyster activity.

animations could be generated on demand for specific areas or times, enabling analysts to scan for patterns that might inspire more detailed ad-hoc analyses of the data.[4]

## 7.3 OBSERVING CHANGES IN RIDER BEHAVIOR AND DEMAND

Chapter 6 demonstrated that the algorithms developed in this thesis are efficient enough to be applied every day on the full set of Oyster, iBus and transaction-count data. A key benefit of acquiring and retaining full populations of passenger data is the ability to track travel behavior over time (Jones et al 1998).

Muhs (2012) uses the outputs of the software developed in this thesis to observe changes in travel behavior and service quality resulting from the reopening of the East London Line (ELL). The former Underground line was extended and integrated into the Overground network, and the study observed riders' responses to the new service as well as the impact that it had on other transport services.

4 Examples of the animation can be viewed at www.jaygordon.net

Following Ng (2011), who used Oyster data to study changes in travel time, route choice, and interchange location (using methods developed by Seaborn [2009]), Muhs studied the effects of the line by observing a panel of 54,000 Oyster cardholders who used the ELL in October of 2011 and whose cards were also active in April of 2010, before the line opened. By conducting various analyses before and after the opening of the line—such as riders' mode choice, travel times, journey frequencies, and the analysis of boarding/alighting/flow profiles on parallel and intersecting bus routes—the study was able to quantify many of the project's effects while testing the predictions of the ELL business case.

In addition to comprehensive studies such as those conducted by Ng and Muhs, the analyses mentioned in the previous sections can be conducted on an ad-hoc basis to get a quick sense of changes in the transport system. Figure 7-7 illustrates the ridershed of the 205X, a special

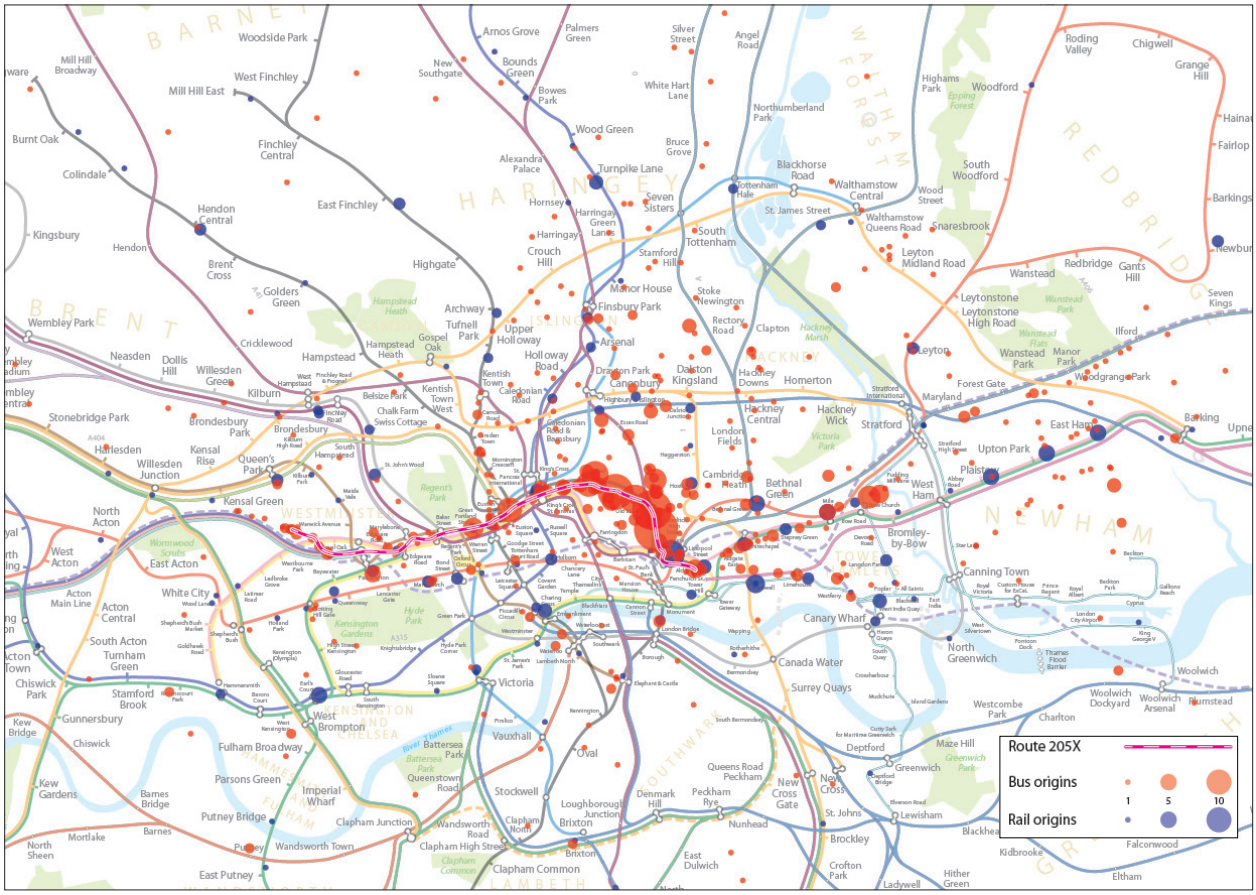


Figure 7-7. First origins of day for riders of route 205X, Sunday, 28 August 2011.

bus service operated to serve the Notting Hill Carnival in the summer of 2011. By mapping the first daily boarding locations of the route's riders, a coarse estimate of their places of residence is obtained.

Similarly, boarding/alighting/flow profiles can be quickly generated to compare travel behavior over time. The profile in Figure 7-8 was generated for route 488, which was extended in 2011 to serve Dalston Junction, the northern terminus of the East London Line. The chart shows the changes in boardings, alightings, and flows, but the ad hoc generation of a matching OD matrix reveals the OD flows that constitute this activity (Figure 7-9).

## 7.4 SUMMARY

The examples presented in this chapter illustrate just a few of the analyses to which the methods of this thesis can be applied. The outputs of the bus origin- and destination-inference processes alone enable several useful types of analysis, and the application of full-journey expansion factors to those data enable bus ridership profiles and OD matrices to be scaled in a way that makes their outputs compatible with other studies.

The interchange-inference process enables the construction of full-journey matrices, which can be used to measure and map various aspects of ridership at a range of spatial and temporal resolutions. Ad-hoc analyses of special events, service closures,[5] or system performance can be undertaken by staff using simple database tools, while large projects such as the East London Line Extension, the 2012 Olympic and Paralympic Games, and Crossrail can be studied in detail longitudinally in order to assess their impacts and inform future projects.

5 AFC data have been used to study passenger behavior during maintenance closures on Chicago's metro system (Mojica 2008).

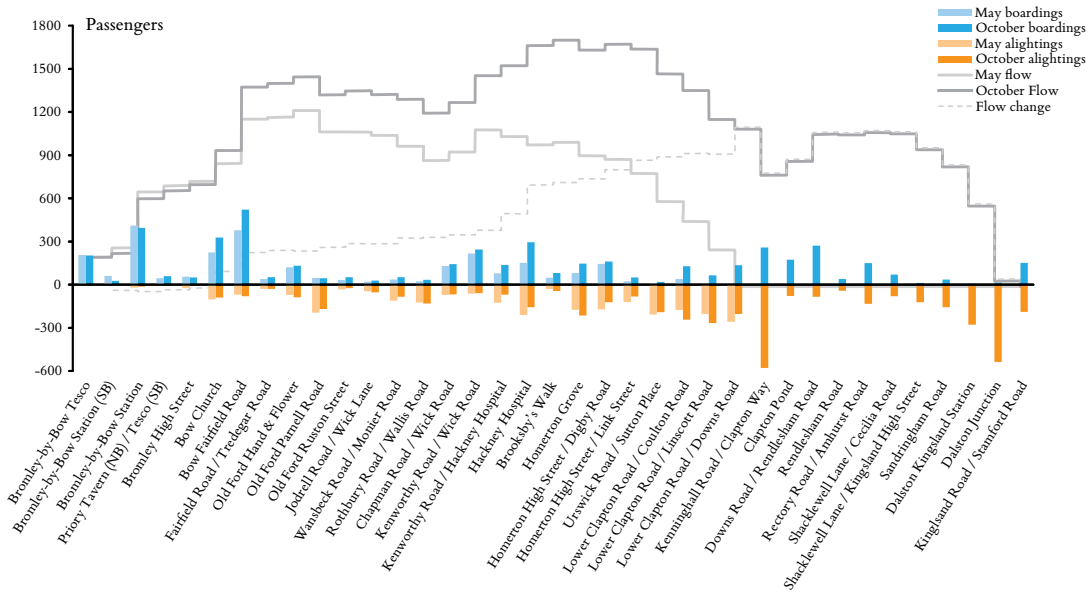


Figure 7-8. Boarding/alighting/flow profile for route 488, before and after its extension to Dalston Junction.

DESTINATION

| ORIGIN | Bromley-by-Bow Tesco | Bromley-by-Bow Station (SB) | Bromley-by-Bow Station | Priory Tavern (NB) / Tesco (SB) | Bromley High Street | Bow Church | Bow Fairfield Road | Fairfield Road / Tredegar Road | Old Ford Hand & Flower | Old Ford Parnell Road | Old Ford Ruston Street | Jodrell Road / Wick Lane | Wansbeck Road / Monier Road | Rothbury Road / Wallis Road | Chapman Road / Wick Road | Kenworthy Road / Wick Road | Kenworthy Road / Hackney Hospital | Hackney Hospital | Brooksby's Walk | Homerton Grove | Homerton High Street / Digby Road | Homerton High Street / Link Street | Urswick Road / Sutton Place | Lower Clapton Road / Coulton Road | Lower Clapton Road / Linscott Road | Lower Clapton Road / Downs Road | Kenninghall Road / Clapton Way | Clapton Pond | Downs Road / Rendlesham Road | Rendlesham Road | Rectory Road / Amhurst Road | Shacklewell Lane / Cecilia Road | Shacklewell Lane / Kingsland High Street | Sandringham Road | Dalston Kingsland Station | Dalston Junction | Kingsland Road / Stamford Road | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bromley-by-Bow Tesco | – | 0 | 8 | 3 | 1 | 13 | 8 | 1 | 1 | 13 | 1 | 1 | 0 | 0 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 3 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 62 |
| Bromley-by-Bow Station (SB) | – | – | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 3 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13 |
| Bromley-by-Bow Station | – | – | – | 0 | 3 | 21 | 27 | 8 | 11 | 22 | 3 | 14 | 31 | 18 | 7 | 7 | 17 | 15 | 6 | 39 | 15 | 3 | 10 | 3 | 4 | 6 | 24 | 3 | 0 | 3 | 0 | 0 | 0 | 0 | 4 | 1 | 0 | 326 |
| Priory Tavern (NB) / Tesco (SB) | – | – | – | – | 0 | 17 | 1 | 0 | 1 | 7 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 1 | 0 | 3 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 45 |
| Bromley High Street | – | – | – | – | – | 10 | 1 | 0 | 3 | 0 | 0 | 4 | 0 | 0 | 0 | 1 | 1 | 0 | 3 | 3 | 6 | 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 41 |
| Bow Church | – | – | – | – | – | – | 13 | 1 | 6 | 27 | 4 | 1 | 4 | 15 | 13 | 11 | 10 | 8 | 3 | 14 | 7 | 4 | 7 | 8 | 4 | 10 | 32 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 8 | 1 | 219 |
| Bow Fairfield Road | – | – | – | – | – | – | – | 7 | 28 | 41 | 7 | 14 | 18 | 8 | 11 | 8 | 13 | 15 | 1 | 7 | 11 | 0 | 6 | 4 | 1 | 11 | 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 3 | 0 | 237 |
| Fairfield Road / Tredegar Road | – | – | – | – | – | – | – | – | 6 | 3 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 3 | 0 | 4 | 1 | 0 | 0 | 3 | 3 | 3 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 3 | 0 | 1 | 44 |
| Old Ford Hand & Flower | – | – | – | – | – | – | – | – | – | 0 | 0 | 0 | 0 | 14 | 0 | 3 | 0 | 1 | 0 | 11 | 1 | 4 | 6 | 4 | 1 | 7 | 18 | 0 | 0 | 0 | 3 | 3 | 0 | 0 | 3 | 3 | 0 | 83 |
| Old Ford Parnell Road | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 0 | 6 | 3 | 0 | 0 | 1 | 0 | 4 | 0 | 1 | 0 | 1 | 0 | 3 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 27 |
| Old Ford Ruston Street | – | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 11 | 1 | 0 | 3 | 3 | 0 | 4 | 1 | 0 | 0 | 0 | 0 | 4 | 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 49 |
| Jodrell Road / Wick Lane | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 3 | 1 | 0 | 0 | 1 | 0 | 0 | 4 | 0 | 1 | 1 | 0 | 1 | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 25 |
| Wansbeck Road / Monier Road | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 0 | 0 | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 24 |
| Rothbury Road / Wallis Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 0 | 0 | 4 | 3 | 1 | 0 | 0 | 0 | 6 | 0 | 1 | 8 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 27 |
| Chapman Road / Wick Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 3 | 1 | 10 | 6 | 10 | 10 | 10 | 3 | 1 | 18 | 15 | 25 | 4 | 1 | 0 | 0 | 0 | 0 | 3 | 1 | 0 | 0 | 122 |
| Kenworthy Road / Wick Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 35 | 6 | 31 | 8 | 6 | 27 | 10 | 24 | 17 | 22 | 8 | 0 | 1 | 1 | 0 | 4 | 0 | 6 | 3 | 4 | 215 |
| Kenworthy Road / Hackney Hospital | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 1 | 3 | 1 | 3 | 7 | 7 | 35 | 6 | 7 | 0 | 0 | 1 | 0 | 0 | 0 | 3 | 4 | 14 | 94 |
| Hackney Hospital | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 6 | 1 | 11 | 35 | 42 | 25 | 17 | 56 | 0 | 6 | 3 | 4 | 1 | 6 | 3 | 8 | 7 | 6 | 239 |
| Brooksby's Walk | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 1 | 1 | 4 | 13 | 17 | 4 | 8 | 4 | 6 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 67 |
| Homerton Grove | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 4 | 8 | 6 | 0 | 4 | 28 | 3 | 7 | 0 | 3 | 3 | 3 | 4 | 1 | 1 | 0 | 80 |
| Homerton High Street / Digby Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 8 | 38 | 14 | 20 | 38 | 1 | 1 | 0 | 6 | 0 | 3 | 6 | 0 | 8 | 0 | 148 |
| Homerton High Street / Link Street | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 3 | 0 | 15 | 7 | 17 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 45 |
| Urswick Road / Sutton Place | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 3 | 3 | 6 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 14 |
| Lower Clapton Road / Coulton Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 3 | 21 | 11 | 7 | 7 | 3 | 1 | 1 | 0 | 1 | 7 | 3 | 70 |
| Lower Clapton Road / Linscott Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 7 | 1 | 10 | 0 | 1 | 3 | 1 | 0 | 0 | 4 | 0 | 28 |
| Lower Clapton Road / Downs Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 7 | 0 | 1 | 0 | 3 | 8 | 4 | 7 | 11 | 3 | 3 | 48 |
| Clapton Pond | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 11 | 14 | 6 | 27 | 13 | 22 | 7 | 8 | 14 | 7 | 129 |
| Kenninghall Road / Clapton Way | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 1 | 10 | 6 | 15 | 10 | 34 | 42 | 27 | 146 |
| Rendlesham Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 6 | 20 | 8 | 13 | 32 | 31 | 110 | 24 | 243 |
| Downs Road / Rendlesham Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 1 | 0 | 3 | 8 | 10 | 4 | 31 |
| Rectory Road / Amhurst Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 4 | 22 | 32 | 58 | 8 | 125 |
| Shacklewell Lane / Cecilia Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 3 | 1 | 18 | 25 | 7 | 55 |
| Shacklewell Lane / Kingsland High Street | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 1 | 4 | 4 | 0 | 10 |
| Sandringham Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 | 24 | 6 | 30 |
| Dalston Kingsland Station | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 4 | 3 | 7 |
| Dalston Junction | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 6 | 6 |
| Kinglsand Road / Stamford Road | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | – | 0 |
| | 0 | 0 | 8 | 3 | 4 | 66 | 51 | 18 | 56 | 112 | 15 | 37 | 55 | 84 | 45 | 37 | 46 | 101 | 31 | 145 | 79 | 52 | 131 | 155 | 171 | 142 | 399 | 51 | 58 | 28 | 84 | 51 | 83 | 105 | 191 | 351 | 128 | 3173 |

Figure 7-9. Origin–destination matrix for the extended Route 488 inbound, 7–10 AM

# Conclusion

8

The methods presented in this thesis have been shown to infer aspects of public-transport passenger activity that are not directly recorded by automated data-collection systems. By providing results that are consistent with those of manual recording techniques, this work provides a means of obtaining similar information at a far lower cost, covering entire systems on all service days rather than relying on small and infrequent samples.

This chapter summarizes the results and performance of these methods and assesses the degrees to which the research objectives were satisfied. Recommendations are then proposed for Transport for London or others who adopt these methods, and suggestions are put forth for future research that could build upon this work.

## 8.1 SUMMARY AND FINDINGS

The bus origin-inference algorithm, previously validated against TfL's BODS survey for a sample of London bus routes (Wang et al 2011), was refined and applied to complete daily sets of AFC and AVL data for ten consecutive weekdays. Boarding locations were inferred for over 96 percent of the system's 6.3 million daily bus transactions when applying a maximum timestamp-matching error of ±5 minutes. Alighting locations and times were inferred for 75.6 percent of bus transactions, using a set of spatial and temporal parameters chosen after evaluating the distributions of inter-stage distances and passenger speeds.

Interchange status was then inferred for 91.6 percent of Greater London's 9.8 million daily journey stages (comprising the aforementioned 6.3 million bus stages plus several of the region's rail modes). Approximately 30 percent of all journey stages were inferred to have been linked to their following transactions, yielding approximately seven million daily full passenger journeys. As with the destination-inference process, parameters were chosen after exploring distributions of temporal and spatial properties of the data set. Results were compared to the London Travel Demand Survey, revealing a difference in journey length (by number of stages) which may indicate sampling bias in the survey or that the interchange-inference parameters were set too conservatively.

Processed Oyster data were used to construct seed matrices of full passenger journeys, including travel spanning multiple public modes, for five consecutive weekdays. Control totals from bus routes and rail stations were adjusted to account for journeys that span multiple time periods, and expansion factors for each full-journey itinerary were then estimated for a set of both full-day and AM-peak-period matrices. When aggregated by route or station, scaled passenger flows matched all control totals to within .005 percent. Passenger flows were then derived for each itinerary's constituent rail journey stages, and the scaled flows were validated against a rail-stage matrix constructed using the well-established iterative proportional fitting method.

All algorithms were implemented in a Java application, which typically performs the origin-, destination-, and interchange-inference processes on a complete daily set of data in less than 20 minutes on a consumer-grade computer. The outputs were then used to construct and expand full-journey origin-interchange-destination matrices, typically in less than 10 minutes. By appending origin, destination, and interchange information to a copy of the original Oyster records and storing full-journey expansion factors separately, information is stored at the resolution of the individual passenger transactions, enabling the analysis of both granular and aggregate travel information.

In addition to being demonstrated for ad-hoc analyses of route- and station-level ridership and service areas, the tools developed in this thesis have been applied to the visualization of daily system-wide passenger activity, a rigorous evaluation of the effects of a recent capital project (Muhs 2012), and the measurement of service quality and reliability (Schil 2012). TfL staff have been testing these tools as a potential replacement for

one or more travel surveys and are in the process of industrializing these methods for use on a daily basis by multiple divisions of the organization.

## 8.2 RECOMMENDATIONS

This research demonstrates that complete populations of automatically collected public-transport data can be processed efficiently enough to support analysis on a daily basis, even for a large system such as London's. The long-term cost of developing the required software and allocating approximately 30 minutes of computing time daily (on a server or merely a workstation) is far less than that of conducting quarterly or annual surveys, which have much smaller sample sizes. It is therefore recommended that transport agencies with AFC and AVL systems consider adopting these methods as a replacement for many of their existing surveys, which would enhance their analytic capabilities while making the majority of their survey budgets available for targeted, supplemental studies.

While the methods in this thesis are largely generalizable to other public transport systems, the implementation developed for this work was designed specifically for TfL and can benefit from several additional enhancements and validations. Since the rate of origin inference directly affects the rates of destination and interchange inference, it is recommended that TfL consider correcting for erroneously defined vehicle trips (as proposed by McCaig and Yip [2010]).[1] Interchange inference similarly affects the reconstruction of full journeys and daily travel histories, and the 91.6 percent inference rate can be improved—beyond the gains from improved origin and destination inference—by obtaining AVL or track data for trams. Doing so would enable the inference of tram alighting locations, and would enable that mode to be included in full-journey analyses.

Attempts should be made to acquire control totals from stations for which they are currently unavailable, such as some newer Overground stations and a number of ungated National Rail stations. Where control

1 Some Oyster bus transactions occur after a vehicle trip's final stop, suggesting that in some cases iBus may have erroneously recorded a vehicle as serving a trip prior to the one that it was actually serving.

totals cannot be obtained, TfL staff should develop a method for estimating default totals. These defaults could be applied to those stations during the extraction of gateline data, or the software could be easily modified to apply them at runtime.

Additional validation is recommended, especially against travel-diary surveys such as LTDS. Respondents are now asked to provide their Oyster card numbers voluntarily, and the matching of survey responses to Oyster activity could help find correlations between the spatial and temporal data provided by Oyster and the qualitative trip-purpose information provided by the survey, thereby enabling the interchange-inference algorithm to more accurately distinguish interchanges from activities.

The analyses of spatial and temporal data conducted in this research—such as the assessment of inter-stage distances and out-of-system speeds—should be conducted at smaller scales, such as during specific time periods, in different geographic areas, or on different types of transport services. The parameters of the three inference processes could then be disaggregated if appropriate, allowing them to be set differently to account for the variations observed across the system and over time.

This research was concerned primarily with the methodology and implementation of the above processes, and has only demonstrated a small sample of its possible applications. The processes should be executed and the results stored daily in a database to enable analysts throughout the organization to perform ad-hoc queries, but reporting tools should also be developed to satisfy the different analytic needs of various groups. For example, bus service planners might benefit from an interface that dynamically constructs boarding/alighting/flow profiles or OD matrices for user-specified routes and time frames, while bus travel speeds could be aggregated by route segment and weighted by inferred passenger flows in order to target the most critical areas for travel-time improvements.[2] Similarly, capital planners might use a tool that maps the origins and destinations associated with user-specified interchange locations in order to assess demand for new facilities and services.

It is recommended that TfL keep complete sets of processed data in a live database for some reasonable duration, such as one or two years, depending on the server infrastructure available. Older data should be

2 Sánchez-Martínez (2012) provides robust analysis tools for bus running time variability and service quality, which could be combined with OD-inferred Oyster data for this purpose.

archived offline, but it is recommended that some subset is retained online, such as one week per month for more recent years and one week per quarter for earlier years. Doing so would allow planners and analysts to explore historical trends when considering service or infrastructure initiatives, and the selection of system-wide data over many points in time could assist in the calibration of the organization's various travel-demand models.

Perhaps the most important determinant of the application and further development of this work will be the replacement of Oyster and other AFC cards with contactless bank cards (RFID-enabled credit and debit cards). TfL and other agencies are presently developing fare payment systems which will allow customers to register bank cards for fare payment, then use the cards as they would AFC cards (Lau 2009, Brakewood and Kocur 2012). Since fare transactions will be processed by banks rather than transport providers, it is critically important that transit operators contractually retain ownership of the transaction data. Failure to do so would render the tools developed in this thesis unusable, or would put agencies in the regrettable position of having to purchase their own data from third parties.

## 8.3 FUTURE RESEARCH

The methods developed in this thesis can enable a number of studies that require large volumes of disaggregate ridership information, and the methods themselves could be improved by synthesizing them with other existing research.

The processing of Oyster journey stages enables the observation of bus passengers' in-vehicle travel time and speeds, travel paths, and interchange locations, but the ability to transfer between rail lines behind the gate obscures customers' paths while making in-vehicle travel time indistinguishable from platform wait time (or from in-system interchange, access, or egress times). Integration of this work with a path-choice model (as demonstrated by Rahbee [2008]) would enable the inference of this information, allowing analyses to be conducted at even finer levels of granularity.

While the assumptions used to infer passenger alightings have been validated against empirical data on several public transport systems (Barry et al 2002 and 2009, Navick and Furth 2002, Zhao et al 2007, Wang et al 2011), it should be expected that not all passengers alight buses at the closest stop to the start of their next journey stage. Combining the nearest-stop rule with observed data from automatic passenger counters (APC) could constrain the possible alighting locations for some bus journey stages, resulting in a more accurate destination-inference algorithm.

The inference of both bus boardings and destinations enables the inference of vehicle load, which can in turn be used to enhance the interchange-inference algorithm. By estimating whether buses are full, or by determining whether a bus is being closely followed by another, the algorithm could more accurately discern whether passengers were waiting for buses or engaged in activities.

This research estimates spatial and temporal information about the in-system and interchange portions of full passenger journeys, but access and egress information could be inferred as well. By using the postcodes of registered Oyster users (Ng 2011, Muhs 2012) to infer access or egress distances (Alshalalfah and Shalaby 2007), a more complete model of passenger journeys can be constructed while providing empirical feedback that could be used to calibrate the distance parameters of the destination- and interchange-inference processes.

Contactless bank cards, if their data are retained by transport agencies, will necessitate changes to the destination- and interchange-inference algorithms. Future research could address the benefits and challenges of these new systems by distinguishing riders from cards (since a card could be used to pay the fare of multiple riders) and by incorporating non-transit purchases if possible, to better discern journey purpose.

A wealth of information could be inferred by applying data mining techniques to several days of processed Oyster records (Fayyad et al 1996, Zhao 2009). Following Morency et al (2007) and Chu et al (2011), the daily patterns of cardholders could be studied over time to infer broad activity types such as work, school, and recreation by analyzing the duration, frequency, and regularity of visits to specific locations or zones. Travel behavior variability could also be studied in this manner—for example, by observing riders' preferences between using the same service repeatedly or choosing from among services (perhaps serving the same corridor) based on arrival time, crowding, traffic congestion or even weather.

Observing a panel of individuals longitudinally could also reveal the patterns in which riders adapt to service changes or switch to new services.

Finally, the methods developed in this thesis can be applied to other public transport systems. The software has been preliminarily tested with data from the Massachusetts Bay Transportation Authority (MBTA), who operate bus, subway, streetcar, and commuter-rail services in the Boston metropolitan area. Such testing reveals the differences between the various inputs and parameters of the two systems. For example, MBTA subway customers tap upon station entry only, necessitating a rail destination-inference process that will be similar in many ways to the process for buses. Furthermore, the configuration and density of Boston's transport system (and underlying land use) is substantially different from that of London, which is likely to require a different set of parameter values. By applying this work to the MBTA and other transit networks, its methods can be made more robust and adaptable, which might ultimately provide a flexible means of enhancing the analytical capacity of transit agencies throughout the world.

# References


Alshalalfah, B.W., and A.S. Shalaby. 2007. "Case Study: Relationship of Walk Access Distance to Transit with Service, Travel, and Personal Characteristics." *Journal of Urban Planning and Development* 133 (2): 114–118.

Anas, Alex. 2007. "A unified theory of consumption, travel, and trip chaining." *Journal of Urban Economics* 62: 162–186.

Bagchi, Mousumi, and Peter White. 2003. "Use of Public Transport Smart Card Data for Understanding Travel Behaviour." *Proceedings of the European Transport Conference*. Strasbourg.

———. 2004. "What role for smart card data from bus systems." *Proceedings of the Institution of Civil Engineers: Municipal Engineer* 157 (March): 39–46.

———. 2005. "The Potential of Public Transport Smart Card Data." *Transport Policy* 12: 464–474.

Barry, James J., Robert Newhouser, Adam Rahbee, and Shermeen Sayeda. 2002. "Origin and Destination Estimation in New York City with Automated Fare System Data." *Transportation Research Record: Journal of the Transportation Research Board* 1817: 183–187.

Barry, James J., Robert Freimer, and Howard Slavin. 2009. "Use of Entry-Only Automatic Fare Collection Data to Estimate Linked Transit Trips in New York City." *Transportation Reserach Record: Journal of the Transportation Research Board* 2112: 53–61.

Ben Akiva, Moshe. 1987. "Methods to Combine Different Data Sources and Estimate Origin–Destination Matrices." *Proceedings of the Tenth International Symposium on Transportation and Traffic Theory*. Nathan H. Gartner and Nigel H.M. Wilson, eds. Cambridge, MA. 459–481.

Ben Akiva, Moshe, P.P. Macke, and P.S. Hsu. 1985. "Alternative Methods to Estimate Route-Level Trip Tables and Expand On-Board Surveys." *Transportation Research Record: Journal of the Transportation Research Board* 1037: 1–11.

Björck, Åke. 1996. *Numerical Methods for Least Squares Problems*. Philadelphia: Society for Industrial and Applied Mathematics.

Brakewood, Candace and George Kocur. 2012. "Modeling Transit Rider Preferences for Contactless Bank Cards as Fare Media: Transport for London and the Chicago Transit Authority." *Transportation Research Record: Journal of the Transportation Research Board* 2216: 100–107.

Chan, Joanne. 2007. *Rail Transit OD Matrix Estimation and Journey Time Reliability Metrics Using Automated Fare Data*. Master's thesis, Massachusetts Institute of Technology.

Chapleau, Robert, Ka Kee Alfred Chu, and Bruno Allard. 2011. "Synthesizing AFC, APC, GPS, and GIS Data to Generate Performance and Travel Demand Indicators for Public Transit." 90th Annual Meeting of the Transportation Research Board, Washington, DC.

Chu, Ka Kee Alfred. 2010. *Leveraging Data from a Smart Card Automatic Fare Collection System for Public Transit Planning*. PhD diss., Université de Montréal.

Chu, Ka Kee Alfred and Robert Chapleau. 2007. "Imputation Techniques for Missing Fields and Implausible Values in Public Transit Smart Card Data." 11th World Conference on Transportation Research, Berkeley.

———. 2008. "Enriching Archived Smart Card Transaction Data for Transit Demand Modeling." *Transportation Research Record: Journal of the Transportation Research Board* 2063: 63–72.

———. 2010. "Augmenting Transit Trip Characterization and Travel Behavior Comprehension." *Transportation Reserach Record* 2183: 29–40.

Clarke, Roger. 2001. "Person Location and Person Tracking: Technologies, Risks, and Policy Implications." *Information Technology & People* 14 (2): 206–231.

Continental Automotive. 2011. "iBus, London: Parser Logic." Neuhausen.

Cui, Alex. 2006. *Bus Passenger Origin–Destination Matrix Estimation Using Automated Data Collection Systems*. Master's thesis, Massachusetts Institute of Technology.

Deming, W. Edwards and Frederick F. Stephan. 1940. "On a Least Squares Adjustment of a Sampled Frequency Table When the expected Marginal Totals are Known." *Annals of Mathematical Statistics* 11 (4): 427–444.

Ehrlich, Joseph E. 2010. *Applications of Automatic Vehicle Location Systems Towards Improving Service Reliability and Operations Planning in London*. Master's thesis, Massachusetts Institute of Technology.

Eom, Jin Ki, and Myoung Joon Sung. 2011. "Analysis of Travel Patterns of the Elderly using Transit Smart Card Data." 90th Annual Meeting of the Transportation Research Board, Washington, DC.

Farzin, Janine M. 2008. "Constructing an Automated Bus O-D Matrix Using Smart Card and GPS Data in São Paulo, Brazil." *Transportation Research Record: Journal of the Transportation Research Board* 2072: 30–37.

Fayyad, Usama, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. "From Data Mining to Knowledge Discovery in Databases." *AI Magazine* (Fall): 37–54.

Frumin, Michael S. 2010. *Automatic Data for Applied Railway Management: Passenger Demand, Service Quality Measurement, and Tactical Planning on the London Overground Network*. Master's thesis, Massachusetts Institute of Technology.

Gordillo, Fabio. 2006. *The Value of Automated Fare Collection Data for Transit Planning: An Example of Rail Transit OD Matrix Estimation*. Master's thesis, Massachusetts Institute of Technology.

Gordon, Jason B. 2011. "A Day in the Life of 3.1 Million Londoners." Last modified December 8. http://www.jaygordon.net.

Guo, Zhan. 2008. *Transfers and Path Choice in Urban Public Transport Systems*. PhD diss., Massachusetts Institute of Technology.

Guo, Zhan and Nigel H.M. Wilson. 2007. "Modeling the Effects of Transit System Transfers on Travel Behavior: The Case of Commuter Rail and Subway in Downtown Boston." *Transportation Research Record: Journal of the Transportation Research Board* 2006: 11–20.

Haberman, Shelby J. 1974. *The Analysis of Frequency Data*. Chicago: University of Chicago Press.

Hardy, Nigel. 2007. "iBus Benefits Realisation Workstream: Method & Progress to Date." 16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services, Stockholm.

Henderson, Liam. 2010. *Origin and Destination Information for the Docklands Light Railway: Collection and Application*. Master's thesis, University of Westminster.

Hine, Julian, Mark Wardman, and Steve Stradling. 2003. "Interchange and Seamless Travel." *Integrated Futures and Transport Choices: UK Transport Policy Beyond the 1998 White Paper and Transport Acts*. Julian Hine and John Preston, eds. Aldershot: Ashgate. 116–131

Hine, Julian and J Scott. 2000. "Seamless, accessible travel: users' views of the public transport journey and interchange." *Transport Policy* 7 (3): 217–226 .

Hofmann, Markus, and Margaret O'Mahony. 2005. "Transfer Journey Identification and Analyses from lectronic Fare Collection Data." *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*. Vienna.

Holton, Richard H. 1958. "The Distinction Between Convenience Goods, Shopping Goods, and Specialty Goods." *Journal of Marketing* 23 (1): 53–56.

Hounsell, N.B., B.P. Shrestha, and A. Wong. 2012. "Data Management and Applications in a World-Leading Bus Fleet." *Transportation Research Part C: Emerging Technologies* 22: 76–87.

Institute of Logistics and Transport. 2000. *Passenger Interchanges: A Practical Way of Achieving Passenger Transport Integration*.

Jain, Nihit. 2011. *Assessing the Impact of Recent Fare Policy Changes on Public Transport Demand in London*. Master's thesis, Massachusetts Institute of Technology.

Jang, Wonjae. 2010. "Travel Time and Transfer Analysis Using Transit Smart Card Data." *Transportation Research Record: Journal of the Transportation Research Board* 2144: 142–149.

Ji, Yuxiong, Rabi G. Mishalani, Mark R McCord, and Prem K. Goel. 2011. "Identifying Homogeneous Periods for bus Route Origin-destination Passenger Flow Patterns Based on Automatic Passenger Count Data." 90th Annual Meeting of the Transportation Research Board, Washington, DC..

Jones, Peter, Karen Lucas, and Julia Bray. 1998. "Methodology and Study Programme for an Impact Assessment of the Effects of the Jubilee Line Extension." *Proceedings of the European Transport Conference 1998: hTransportation Planning Methods Vol II*: 123–136.

Knuth, Donald E. 1973. *The Art of Computer Programming*. Vol. 3. Reading, MA: Addison Wesley.

Krizek, Kevin J. 2003. "Neighborhood services, trip purpose, and tour-based travel." *Transportation* 30: 387–410.

Kuhnimhof, Tobias, and Volker Wassmuth. 2002. "Do You Go to the Movies on your Lunch Break? Trip-Context Data-Based Modeling of Activities." *Transportation Research Record: Journal of the Transportation Research Board* 1807: 34–42.

Lau, Peter S.C. 2009. *Developing a Contactless Bankcard Fare Engine for Transport for London*. Master's thesis, Massachusetts Institute of Technology.

Lu, Alex, and Alla Reddy. 2011. "An Algorithim to Measure Daily Bus Passenger Miles using Electronic Farebox Data for National Transit Database (NTD) Section 15 Reporting." 90th Annual Meeting of the Transportation Research Board, Washington, DC.

McCaig, Ewen, and Mandy Yip. 2010. "Technical note: A-BODS Stage 2." Transport for London.

McCord, Mark R., Rabi G. Mishalani, Prem Goel, and Brandon Strohl. 2010. "Iterative Proportional Fitting Procedure to Determine Bus Route Passenger Origin-Destination Flows." *Transportation Research Record: Journal of the Transportation Research Board* 2145: 59–65.

Metropolitan Transportatino Commission. 2003. "Trip Linking Procedures: Working Paper #2: Bay Area Travel Survey 2000." Oakland.

Mishalani, Rabi G., Yuxiong Ji, and Mark R. McCord. 2011. "Empirical Evaluation of the Effect of Onboard Survey Sample Size on Transit Bus Route Passenger OD Flow Matrix Estimation Using APC Data." 90th Annual Meeting of the Transportation Research Board, Washington, DC.

Mojica, Carlos H. 2008. *Examining Changes in Transit Passenger Travel Behavior through a Smart Card Activity Analysis*. Master's thesis, Massachusetts Institute of Technology.

Morency, Catherine, Martin Trépanier, and Bruno Agard. 2007. "Measuring transit use variability with smart-card data." *Transit Policy* 14: 193–203.

Muhs, Kevin. 2012. *Using Automatically Collected Data to Infer Travel Behavior: A Case Study of the East London Line Extension*. Master's thesis, Massachusetts Institute of Technology.

Munizaga, Marcela, Carolina Palma, and Daniel Fischer. 2011. "Estimation of a Disaggregate Multimodal Public Transport OD Matrix from Passive Smart Card Data from Santiago, Chile." 90th Annual Meeting of the Transportation Research Board, Washington, DC.

Navick, David S, and Peter G. Furth. 2002. "Estimating Passenger Miles, Oringi-Destination Patterns, and Loads with Location-Stamped Farebox Data." *Transportation Research Record: Journal of the Transportation Research Board* 1799: 107–113.

Ng, Albert. 2011. *Use of Automatically Collected Data for the Preliminary Impact Analysis of the East London Line Extension*. Master's thesis, Massachusetts Institute of Technology.

Okamura, T., J. Zhang, and F. Akimasa. 2004. "Finding Behavioral Rules of Urban Public Transport Passengers by Using Boarding Records of Integrated Stored Fare Card System." 10th World Conference on Transport Research. Istanbul.

Park, Jin Young, Dong-Jun Kim, and Yongtaek Lim. 2008. "Use of Smart Card Data to Define Public Transit Use in Seoul, South Korea." *Transportation Reserach Record* 2063: 3–9.

Paul, Elizabeth C. 2010. *Estimating Train Passenger Load from Automated Data Systems: Application to London Underground*. Master's thesis, Massachusetts Institute of Technology.

Pukelsheim, Freidreich. 2012. "An $L_1$-Analysis of the Iterative Proportional Fitting Procedure." Preprints: Institut für Mathematik der Universität Augsburg.

Rahbee, Adam B. 2008. "Farecard Passenger Flow Model at Chicago Transit Authority, Illinois." *Transportation Research Record: Journal of the Transportation Research Board* 2072: 3–9.

Raveau, Sebastián, Juan Carlos Muñoz, and Louis de Grange. 2010. "A Topological Route Choice Model for Metro." *Transportation Research* A 45: 138–147.

Robinson, Steve. 2007. "A brief introduction to London Buses nodes, sequences, and schedules." Ver. 00d. Transport for London, Surface Transport.

———. 2010. "161: Interface between iBus and CSI: Static Data." Ver. 08. Transport for London, Surface Transport.

Robinson, Steve and Mauro Manela. 2012. "Automatic Identification of Vehicles with Faulty AVLC Units in the London Buses' iBus System." 91st Annual Meeting of the Transportation Research Board, Washington, DC.

Sánchez-Martínez, Gabriel E. 2012. *Running Time Variability, Curtailments, and Resource Allocation: A Data-Driven Analysis of High-Frequency Bus Operations*. Master's thesis, Massachusetts Institute of Technology.

Schil, Mickaël. 2012. *Measuring Travel Time Reliability in London Using Automated Data Collection Systems*. Master's thesis, Massachusetts Institute of Technology.

Seaborn, Catherine W., John P. Attanucci, and Nigel H.M. Wilson. 2009. "Analyzing Multimodal Public Trasport Journeys in London with Smart Card Fare Payment Data." *Transportation Research Record: Journal of the Transportation Research Board* 2121: 55–62.

Simon, Jesse. 2010. "Origin/Destination Applications from Smart Card Data." Los Angeles County Metropolitan Transportation Authority.

Simon, Jesse, and Peter G. Furth. 1985. "Generating a Bus Route O-D Matrix from On-Off Data." *Journal of Transportation Engineering* 111 (6): 583–593.

Stopher, Peter R. 2008. "The Travel Survey Toolkit: Where To From Here." International Conference on Travel Survey Methods, Annecy.

Thill, Jean-Claude, and Isabelle Thomas. 1987. "Toward Conceptualizing Trip-Chaining Behavior: A Review." *Geographical Analysis* 19 (1): 1–17.

Timmermans, Harry, Peter van der Waerden, Mario Alves, John Polak, Scott Ellis, Andrew S. Harvey, Shigeyuki Kurose, and Rianne Zandee. 2002. "Time allocation in urban and transport settings: an international, inter-urban perspective." *Transport Policy* 9: 79–93.

Transport for London. 2001. *Intermodal Transport Interchange for London: Best Practice Guidelines*.

———. 2002. *Interchange Plan: Improving Interchange in London*.

———. 2006. "East London Line Business Case Update."

———. 2006. "Bus Priority at Traffic Signals Keeps London's Buses Moving: Selective Vehicle Detection (SVD)."

———. 2009. *Interchange Best Practice Guidelines*.

———. 2010. "Transport for London: Factsheet." http://www.tfl.gov.uk/assets/downloads/corporate/transport-for-london-factsheet%281%29.pdf

———. 2011. "Annual Report and Statement of Accounts: 2010/11."

———. 2012. "." *Transport for London*. http://www.tfl.gov.uk/corporate/modesoftransport/londonbuses/1554.aspx

Trépanier, Martin, Nicolas Tranchant, and Robert Chapleau. 2007. "Individual Trip Destination Estimation in a Transit Smart Card Automated Fare Collection System." *Journal of Intelligent Transportation Systems* 11 (1): 1–14.

Uniman, David L., John P. Attanucci, Rabi G. Mishalani, and Nigel H.M. Wilson. 2010. "Service Reliability Measurement Using Automated Fare Card Data: Application to the London Underground." *Transportation Research Record: Journal of the Transportation Research Board* 2143: 92–99.

Utsunomiya, Mariko, John Attanucci, and Nigel H.M. Wilson. 2006. "Potential Uses of Transit Smart Card Registration and Transaction Data to Improve Transit Planning." *Transportation Research Record: Journal of the Transportation Research Board* 1971: 119–126.

Wardman, M. and J. Hine. 2000. "Costs of lnterchange: A Review of the Literature." Working Paper 546, Institute for Transport Studies, University of Leeds.

Wong, K.I., S.C. Wong, C.O. Tong, W.H.K. Lam, H.K. Lo, H. Yang, and H.P. Lo. 2005. "Estimation of Origin–Destination Matrices for a Multimodal Public Transit Network." *Journal of Advanced Transportation* 39 (2): 139–168.

Wang, Wei. 2010. *Bus Passenger Origin–Destination Estimation and Travel Behavior Using Automated Data Collection Systems in London, UK*. Master's thesis, Massachusetts Institute of Technology.

Wang, Wei., John P. Attanucci, and Nigel H.M. Wilson. 2011. "Bus Passenger Origin–Destination Estimation and Related Analyses Using Automated Data Collection Systems." *Journal of Public Transportation* 14 (4): 131–150.

Wilson, Nigel H.M., John P. Attanucci, Joanne Chan, and Alex Cui. 2008. "The use of automated data collection systems to improve public transport performance." *Proceedings of the 10th International Conference on Application of Advanced Technologies in Transportation*. Athens.

Zhao, Jinhua. 2004. *The Planning and Analysis Implications of Automated Data Collection Systems: Rail Transit OD Matrix Inference and Path Choice Modeling Examples*. Master's thesis, Massachusetts Institute of Technology.

———. 2009. *Preference Accommodating and Preference Shaping: Incorporating Traveler Preferences into Transportation Planning*. PhD diss., Massachusetts Institute of Technology.

Zhao, Jinhua, Adam Rahbee, and Nigel H.M. Wilson. 2007. "Estimating a Rail Passenger Trip Origin–Destination Matrix Using Automatic Data Collection Systems." *Computer-Aided Civil and Infrastructure Engineering* 22 (5): 376–387.